RedHat/CentOS/Oracle 6 Data Security Standard Mapping - PCI v3.1

Date:	4/22/16 10:42 AM
Show descendant test groups:	Yes
Display criteria at end:	No
Show full details:	Yes
Weight:	All
Test Severity range:	All
Has remediator:	Not applied
Tests:	RHEL 6 Data Security Standard Mapping - PCI v3.1

RHEL 6 Data Security Standard Mapping - PCI v3.1

Nodes	SF - Redhat/CentOS/Oracle 6 - Policy

Requirement 1 Install and Maintain a Firewall Configuration to Protect Cardholder Data

Firewalls are devices that control computer traffic allowed between an entity's networks (internal) and un trusted networks (external), as well as traffic into and out of more sensitive areas within an entity's internal trusted networks. The cardholder data environment is an example of a more sensitive area within an entit y's trusted network.
A firewall examines all network traffic and blocks those transmissions that do not meet the specified secu rity criteria.
All systems must be protected from unauthorized access from untrusted networks, whether entering the system via the Internet as e-commerce, employee Internet access through desktop browsers, employee e-mail access, dedicated connections such as business-to-business connections, via wireless networks, or via other sources. Often, seemingly insignificant paths to and from untrusted networks can provide unpro tected pathways into key systems. Firewalls are a key protection mechanism for any computer network.

1.2 Firewall Configuration

Build firewall and router configurations that restrict connections between untrusted networks and any sys tem components in the cardholder data environment.
Note: An "untrusted network" is any network that is external to the networks belonging to the entity under review, and/or which is out of the entity's ability to control or manage.

1.2.1 Allow Only Necessary Traffic

Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.

1.2.1.1 Verify That IPtables Is Enabled

Verify That IPtables Is Enabled

Description	IPtables is an application that allows a system administrator to configure the IPv4 tables, chains and rules provided by the Linux kernel firewall. It is recommended that IPtables is Enabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Excluded Nodes	Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "service status"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*iptables[\ \t]+.*\bon\b.*$\n(?!(?:^.*$\n)*^([\ \t]*Firewall[\ \t]+is[\ \t]+stopped|[\ \t]*iptables:[\ \t]+Firewall[\ \t]+is[\ \t]+not[\ \t]+running))/ (Flags:Multiline,Comments mode)
iptables Service Status Exists

Remediation

To remediate failure of this policy test, enable the iptables service.

Enabling the iptables service:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/service iptables start** command to start the **iptables** service.
3. Run the **/sbin/chkconfig iptables on** command to keep the **iptables** service enabled in the next reboot.

For further details, please run the command **man iptables** to read man page.

1.3 Prohibit Direct Public Access

Prohibit direct public access between the Internet and any system component in the cardholder data environment.

1.3.4 Anti-spoofing Measures

Implement anti-spoofing measures to detect and block forged source IP addresses from entering the network.
(For example, block traffic originating from the Internet with an internal source address.)

1.3.4.1 Verify That the Systems Local IPv4 Firewall Implements a Deny-all, Allow-by-exception Policy for Inbound Packets

Verify That the Systems Local IPv4 Firewall Implements a Deny-all, Allow-by-exception Policy for Inbound Packets

Description	This test verifies that the systems local IPv4 firewall implement a deny-all, allow-by-exception policy for inbound packets. In "iptables" the default policy is applied only after all the applicable rules in the table are examined for a match. Setting the default policy to "DROP" implements proper design for a firewall, i.e. any packets which are not explicitly permitted should not be accepted.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 6
Element	Equals "/etc/sysconfig/iptables"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^:[\ \t]*INPUT[\ \t]+DROP[\ \t].*$/ (Flags:Multiline,Comments mode) INPUT Chain Is Set to DROP Exists
Remediation	To remediate failure of this policy test, configure INPUT chain to DROP. **Configuring INPUT chain to DROP:** 1. Become superuser or assume an equivalent role. 2. Run the **/sbin/iptables -P INPUT DROP** command to drop incomming package. 3. Run the **service iptables save** command to add the settings to the **/etc/sysconfig/iptables** file. 4. Run the command **service iptables restart** to apply changes. For further details, please run the command **man iptables** to read man page. **Note**: Do not set the default policy to drop incomming packets to the system without setting allow-by-exception policy for needed inbound packages.

1.3.4.2 Verify That the System's Local Firewall Implements a Deny-all, Allow-by-exception Policy for Forwarded Packets

Verify That the System's Local Firewall Implements a Deny-all, Allow-by-exception Policy for Forwarded Packets

Description

This test verifies that the system's local firewall must implement a deny-all, allow-by-exception policy for forwarded packets. In "iptables" the default policy is applied only after all the applicable rules in the table are examined for a match. Setting the default policy to "DROP" implements proper design for a firewall, i.e. any packets which are not explicitly permitted should not be accepted.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "/etc/sysconfig/iptables"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^:[\ \t]*FORWARD[\ \t]+DROP[\ \t]+.*(?:$|\#.*)/ (Flags:Multiline,Comments mode)
DROP Forwarded Packets Exists

Remediation

To remediate failure of this policy test, configure the firewall to drop forwarded packages by default.

Configure the firewall to drop forwarded packages by default:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/iptables -P FORWARD DROP** command to drop forwarded package.
3. Run the **service iptables save** command to add the settings to the **/etc/sysconfig/iptables** file.
4. Run the command **service iptables restart** to apply changes.

For further details, please run the command **man iptables** to read man page.

Requirement 2 Do Not Use Vendor-supplied Defaults for System Passwords and Other Security Parameters

Malicious individuals (external and internal to an entity) often use vendor default passwords and other vendor default settings to compromise systems. These passwords and settings are well known by hacker communities and are easily determined via public information.

2.1 Change Vendor-supplied Defaults

Always change vendor-supplied defaults and remove or disable unnecessary default accounts before installing a system on the network.
This applies to ALL default passwords, including but not limited to those used by operating systems, software that provides security services, application and system accounts, point-of-sale (POS) terminals, Simple Network Management Protocol (SNMP) community strings, etc.).

2.1.0 Change Non-Wireless Vendor Defaults

Always change vendor-supplied defaults and remove or disable unnecessary default accounts before installing a system on the network.
This applies to ALL default passwords, including but not limited to those used by operating systems, software that provides security services, application and system accounts, point-of-sale (POS) terminals, Simple Network Management Protocol (SNMP) community strings, etc.).

2.1.0.1 Verify That Default Login Shell for System Accounts Is Set to /sbin/nologin

Verify That Default Login Shell for System Accounts Is Set to /sbin/nologin

<table>
<tr><td><b>Description</b></td><td>This test verifies that default login shell for system accounts is set to /sbin/nologin. It is important to make sure that accounts that are not being used by regular users are locked to prevent them from being used to provide an interactive shell and it is also recommended that the shell field in the password file be set to /sbin/nologin. This prevents the account from potentially being used to run any commands.</td></tr>
<tr><td><b>Severity</b></td><td>0</td></tr>
<tr><td><b>Weight</b></td><td>5</td></tr>
<tr><td><b>Type</b></td><td>Content Test</td></tr>
<tr><td><b>Rules</b></td><td>Block System Accounts</td></tr>
<tr><td><b>Excluded Nodes</b></td><td>Red Hat Enterprise Linux Server 7
Red Hat Enterprise Linux Server 6
Red Hat Enterprise Linux Server 5</td></tr>
<tr><td><b>Element</b></td><td>Equals "System Accounts"</td></tr>
<tr><td><b>Version conditions</b></td><td>If an element version has no content, the condition should:Pass
Regular expression: /^Username=(?!(?:sync|shutdown|halt)\b)\S+[\ \t]+Id=(?:\d{0,2}|[1-4]\d{2})[\ \t]+Shell=(?!/sbin/nologin\b).*/ (Flags:Multiline,Comments mode)
System Account Setting Deviation Does not exist</td></tr>
<tr><td><b>Remediation</b></td><td>To remediate failure of this policy test, set default login shell for the system accounts to /sbin/nologin.
<b>Setting the default login shell for the system accounts to /sbin/nologin</b>:
1. Become superuser or assume an equivalent role.
2. Run the script:
<b>/bin/awk -F: '0+$3 < 500 && $1 !~ /^[[:space:]]*(#.*|root|sync|shutdown|halt|\+.*)$/ && $7 !~ /^\/sbin\/nologin$/ {print $1}' /etc/passwd 2>/dev/null</b>
to list all the system accounts that do not have <b>/sbin/nologin</b> as the default login shell.
3. For each account listed in step 2, run command the <b>usermod -s /sbin/nologin <account_name></b> command to set default login shell for the account to <b>/sbin/nologin</b>.
For further details, please run the command <b>man usermod</b> to read man page.</td></tr>
</table>

2.1.0.2 Verify That Default Group ID for root Account Is 0

Verify That Default Group ID for root Account Is 0

Description	Using GID 0 for the root account helps prevent root-owned files from accidentally becoming accessible to non-privileged users.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/passwd"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*root:[^:]+:\d+:(\d+):.*$/ (Flags:Multiline,Comments mode) Default GroupID for root Equals 0
Remediation	To remediate failure of this policy test, set the default GID for root to 0. **Setting the default GID for root to 0**: 1. Become superuser or assume an equivalent role. 2. Run the **usermod -g 0 root** command to set the default **GID** for **root** to 0. For further details, please run the command **man usermod** to read man page.

2.1.0.3 Verify That sshd_config Contains a Banner for Network Access

Verify That sshd_config Contains a Banner for Network Access

Description

This test verifies that the SSH server is configured to display a login banner message when it is accessed. The presence of a login banner is useful when prosecuting trespassers of the computer system. Additionally, it can have the effect of obfuscating important operating system information.

Severity

0

Weight

5

Type

Content Test

Rules

Verify Banner Message in /etc/ssh/sshd_config

Element

Equals "Banner Message"

Version conditions

Action if missing:Fail
Banner Entry Equals "Exist"

Remediation

To remediate failure of this policy test, configure the SSH daemon to use safe defaults for the client and server by setting a banner message for use during SSH logins.

Configuring the SSH Server to use a banner:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Find the line

   Banner <banner_file>

   where **<banner_file>** is **/etc/issue.net** or **/etc/issue**
4. Uncomment that line or add if not found and save the file.
5. Run the following command to create a banner message in the **<banner_file>** file.

   echo "<banner_message>" >> <banner_file>

   where **<banner_message>** is a message that you would like any user who connects to your SSH service to see, as example: ***"Authorized uses only. All activity may be monitored and reported"*** .
6. Run the **service sshd restart** commands to restart the **sshd** service.

Note: If a banner message existed in the **<banner_file>** file, you needn't execute step 5.

For further details, please run the command **man sshd_config** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
#/bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
BannerLine="Authorized uses only. All activity may be monitored
 and reported."

# Script Functions
AddLine(){
    FileName=$1; Line=$2
    AddLog=`(/bin/echo "$Line" >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        if [ -n "$SuccMsg" ]; then
            /bin/echo "FAILURE-7001: Could not add [$Line] line"\
                "to [$FileName] file"
            SuccMsg=`/bin/echo -e "$SuccMsg" | /bin/sed '$d'`
            /bin/echo -e "$SuccMsg"
            exit 7001
        fi
        /bin/echo "FAILURE-6001: Could not add [$Line] line"\
            "to [$FileName] file"
        exit 6001
    else
        if [ -z "$SuccMsg" ]; then
            SuccessCode=6003
        else
            SuccessCode=7001
        fi
        SuccMsg=$SuccMsg"[$Line] line added to [$FileName] file
\n"
    fi
}

# Issue commands to remediate files
if [ ! -e "$FileName" ]; then
    /bin/echo "FAILURE-1002: [$FileName] file/directory does not
 exist"
    exit 1002
fi

BannerFile=`/bin/awk 'tolower($1) ~ /^banner$/{print $2}'
 "$FileName"\
    2>/dev/null`

if [ -f "$BannerFile" -o "$BannerFile" == "/etc/issue.net" ];
 then
    AddLine "$BannerFile" "$BannerLine"
else
    # Remediate /etc/ssh/sshd_config
    if [ -e "$FileName" ]; then
        BaseName=`/bin/basename "$FileName" 2>/dev/null`
        DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
        FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
        if [ ! -d "$FullPath" ]; then
            CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
            if [ -n "$CreateLog" ]; then
               /bin/echo "FAILURE-1003: Could not create"\
                   "[$FullPath] file/directory"
                exit 1003
            fi
        fi
        BackupName="$FullPath/${BaseName}.tecopy"
        CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
        if [ -n "$CopyLog" ]; then
            /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
            exit 1007
        fi
    fi

    IsExisted=`/bin/egrep -i "^[[:space:]]*banner[[:space:]]"\
        "$FileName" 2>/dev/null`

    if [ -z "$IsExisted" ]; then
        AddLine "$FileName" "Banner /etc/issue.net"
    else
        UpdateLog=`(/bin/awk -F"#" 'BEGIN{OFS="#"}
            tolower($1) ~ /^[[:space:]]*banner\>/{
            $1 = "Banner /etc/issue.net"
        }{print}' "$BackupName" > "$FileName") 2>&1`
        if [ -n "$UpdateLog" ]; then
            /bin/echo "FAILURE-7001: Could not update the
 argument of [Banner]"\
                "keyword to [/etc/issue.net] in [$FileName] file"
            /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
            exit 7001
        else
            SuccMsg=$SuccMsg"Argument of [Banner] keyword updated
 to"
            SuccMsg=$SuccMsg" [/etc/issue.net] in [$FileName]
 file\n"
            SuccessCode="7001"
        fi
    fi
    FileName="/etc/issue.net"
```

Post Remediation Category	`Other`
Remediated Elements	`/etc/ssh/sshd_config` `/etc/issue.net`
Post Remediation Steps	To complete this remediation: 1. Become superuser or assume an equivalent role. 2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

2.1.0.4 Verify That System Accounts Are Locked

Verify That System Accounts Are Locked

Description

This test verifies that system accounts are locked. It is important to make sure that accounts that are not being used by regular users are locked to prevent them from being used to provide an interactive shell and it is also recommended that the shell field in the password file be set to /sbin/nologin. This prevents the account from potentially being used to run any commands.

Severity

0

Weight

5

Type

Content Test

Rules

Block System Accounts

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

CentOS 6

CentOS Linux release 7.2.1511

Element

Equals "System Accounts"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^Username=\S+[\ \t]+Id=(?:\d{0,2}|[1-4]\d{2})[\ \t]+Shell=\S+[\ \t]+Account_locked=(?!LK\b).*/ (Flags:Multiline,Comments mode)
System Account Setting Deviation Does not exist

Remediation

To remediate failure of this policy test, lock the system accounts.

Locking the system accounts:

1. Become superuser or assume an equivalent role.
2. Run the script:

```
SystemAccounts=`/bin/awk -F: '0+$3 < 500 && $1 !~ /^[[:space:]]*(#.*|root|\+.*)$/ {print $1}' /etc/passwd 2>/dev/null`; for SystemAccount in $SystemAccounts; do Account_locked=`/usr/bin/passwd -S $SystemAccount 2>/dev/null | /bin/awk '$2 !~ /^LK$/{print $2}'`; if [ -n "$Account_locked" ]; then /bin/echo "$SystemAccount"; fi; done
```

   to list all the system accounts that are not locked.
3. For each account listed in step 2, run command the **usermod -L <account_name>** command to lock the account.

For further details, please run the command **man usermod** to read man page.

2.1.0.5 Verify That There Are No Accounts with Empty Password Fields

Verify That There Are No Accounts with Empty Password Fields

Description	This test determines if any individual accounts listed in /etc/shadow have empty passwords. All accounts should have strong passwords or the account should be locked.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/shadow"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^:\#]+::/ (Flags:Multiline,Case insensitive,Comments mode) Empty Password Accounts Does not exist
Remediation	To remediate failure of this policy test, set the passwords or lock the accounts. **Setting the passwords or locking the accounts**: 1. Become superuser or assume an equivalent role. 2. Run the **awk -F: '($2 == "") { print $1 }' /etc/shadow** command to print the accounts with empty passwords. 3. Run the **passwd <user_name>** command to set the password or run the **passwd <user_name> -l** command to lock the account. For further details, please run the command **man 5 shadow** to read man page.

2.1.0.6 Verify Warning Banners in /etc/issue Do Not Contain OS Information

Verify Warning Banners in /etc/issue Do Not Contain OS Information

Description

This test determines if the banner configured in /etc/issue contains information that would indicate the type of operating system.
Removal of operating system information from login banners helps to prevent attackers from targeting OS vulnerabilities.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/issue"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /\\m|\\r|\\s|\\v/
System Information Does not exist

Remediation

To remediate failure of this policy test, configure the banners to create warnings for network and physical access services in the /etc/issue file.

Configuring the banners for console access in the /etc/issue file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/issue** file.
3. Edit the file to include warning messages for network and physical access services.
4. Remove system information such as: **\m \r \s \v** from the above file if they are present and save the file.

For further details, please run the command **man issue** to read man page.

2.1.0.7 Verify Warning Banners in /etc/motd Do Not Contain OS Information

Verify Warning Banners in /etc/motd Do Not Contain OS Information

Description

This test determines if the banner configured in /etc/motd contains information that would indicate the type of operating system.
Removal of operating system information from login banners helps to prevent attackers from targeting OS vulnerabilities.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/motd"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /\\m|\\r|\\s|\\v/
System Information Does not exist

Remediation

To remediate failure of this policy test, configure the banners to create warnings for network and physical access services in the /etc/motd file.

Configuring the banners for console access in the /etc/motd file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/motd** file.
3. Edit the file to include warning messages for network and physical access services.
4. Remove system information such as: **\m \r \s \v** from the above file if they are present and save the file.

For further details, please run the command **man motd** to read man page.

2.1.0.8 Verify Warning Banners in /etc/issue.net Do Not Contain OS Information

Verify Warning Banners in /etc/issue.net Do Not Contain OS Information

Description

This test determines if the banner configured in /etc/issue.net contains information that would indicate the type of operating system.
Removal of operating system information from login banners helps to prevent attackers from targeting OS vulnerabilities.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/issue.net"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /\\m|\\r|\\s|\\v/
System Information Does not exist

Remediation

To remediate failure of this policy test, configure the banners to create warnings for net work and physical access services in the /etc/issue.net file.

Configuring the banners for console access in the /etc/issue.net file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/issue.net** file.
3. Remove system information such as: **\m \r \s \v** from the above file if they are present and save it.

2.2 Develop Configuration Standards For All System Components

Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards.
Sources of industry-accepted system hardening standards may include, but are not limited to:
- Center for Internet Security (CIS)
- International Organization for Standardization (ISO)
- SysAdmin Audit Network Security (SANS)
- National Institute of Standards Technology (NIST)

2.2.2 Disable Unnecessary Services and Protocols

Enable only necessary services, protocols, daemons, etc., as required for the function of the system.

2.2.2. 1 Verify That the Berkeley rsh-server (rsh, rlogin, rcp) Package Is Removed

Verify That the Berkeley rsh-server (rsh, rlogin, rcp) Package Is Removed

Description	The Berkeley rsh-server (rsh, rlogin, rcp) package contains legacy services that ex change credentials in clear-text. It is recommended that The Berkeley rsh-server (rsh, rlogin, rcp) package is removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*rsh-server-\d.*$ / (Flags:Multiline,Comments mode) rsh-server Does not exist
Remediation	To remediate failure of this policy test, erase rsh-server package. **Erasing rsh-server package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase rsh-server** command to remove **rsh-server** package. For further details, please run the command **man yum** to read man page.

2.2.2. 2 Verify That DHCP Server Is Not Installed on the System

Verify That DHCP Server Is Not Installed on the System

Description	This test verifies that DHCP server is not installed on the system. Unless a server is specifically set up to act as a DHCP server, it is recommended that this service be deleted to reduce the potential attack surface.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*dhcp-\d.*$/ (Flags:Multiline,Comments mode) DHCP Server Does not exist
Remediation	To remediate failure of this policy test, remove DHCP server. **Removing DHCP server:** 1. Become superuser or assume an equivalent role. 2. Run **yum erase dhcp** to remove DHCP server. For further details, please run the command **man yum** to read man page.

2.2.2. 3 Verify That the SETroubleshoot Package Is Removed

Verify That the SETroubleshoot Package Is Removed

Description	The SETroubleshoot service notifies desktop users of SELinux denials through a user-friendly interface. The service provides important information around configuration errors, unauthorized intrusions, and other potential errors. It is recommended that the SETroubleshoot package is removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*setroubleshoot-\d.*$/ (Flags:Multiline,Comments mode) setroubleshoot Does not exist
Remediation	To remediate failure of this policy test, erase the SETroubleshoot package. **Erasing the SETroubleshoot package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase setroubleshoot** command to remove the **SETroubleshoot** package. For further details, please run the command **man yum** to read man page.

2.2.2. 4 Verify That the mcstrans Package Is Removed

Verify That the mcstrans Package Is Removed

Description	The mcstransd daemon provides category label information to client processes requesting information. The label translations are defined in /etc/selinux/targeted/setrans.conf Since this service is not used very often, disable it to reduce the amount of potentially vulnerable code running on the system.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*mcstrans-\d.*$/ (Flags:Multiline,Comments mode) mcstrans Does not exist
Remediation	To remediate failure of this policy test, erase the mcstrans package. **Erasing the mcstrans package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase mcstrans** command to remove the **mcstrans** package. For further details, please run the command **man yum** to read man page.

2.2.2. 5 Verify That telnet Is Disabled

Verify That telnet Is Disabled

Description	This test verifies that telnet is disabled. Telnet uses an unencrypted network protocol, which means data from the login session can be stolen by eavesdroppers on the net work, and also that the session can be hacked by outsiders to gain access to the remote system.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Excluded Nodes	Red Hat Enterprise Linux Server 6
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*telnet(.*)$/ (Flags:Multiline,Comments mode) Telnet Service Status Excludes "on"
Remediation	To remediate failure of this policy test, disable the telnet service. **Disabling the telnet service:** 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list telnet** command to check the status of the service. 3. Disable the **telnet** service using the **chkconfig telnet off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2. 6 Verify That the telnet-server Package Is Removed

Verify That the telnet-server Package Is Removed

Description	The telnet-server package contains the telnetd daemon, which accepts connections from users from other systems via the telnet protocol. The telnet protocol is insecure and unencrypted. The use of an unencrypted transmission medium could allow a user with access to sniff network traffic the ability to steal credentials. It is recommended that The telnet-server package is removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Excluded Nodes	Red Hat Enterprise Linux Server 5
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*telnet-server-\d.*$/ (Flags:Multiline,Comments mode) telnet-server Does not exist
Remediation	To remediate failure of this policy test, erase telnet-server package. **Erasing telnet-server package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase telnet-server** command to remove **telnet-server** package. For further details, please run the command **man yum** to read man page.

2.2.2. 7 Verify That the telnet Package Is Removed

Verify That the telnet Package Is Removed

Description	The telnet package contains the telnet client, which allows users to start connections to other systems via the telnet protocol. The telnet protocol is insecure and unencrypted. The use of an unencrypted transmission medium could allow a user with access to sniff network traffic the ability to steal credentials. It is recommended that The telnet package is removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*telnet-\d.*$/ (Flags:Multiline,Comments mode) telnet Does not exist
Remediation	To remediate failure of this policy test, erase the telnet package. **Erasing the telnet package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase telnet** command to remove **telnet** package. For further details, please run the command **man yum** to read man page.

2.2.2. 8 Verify That the tftp Service Is Disabled

Verify That the tftp Service Is Disabled

Description

This test checks that the tftp service is disabled.
Because SSH provides a mechanism for both a secure login and a secure file transfer, it is not necessary for the insecure services that are supervised by xinetd to be running.
NOTE: It may be necessary for an organization to have one or more xinetd service running. If this is the case, only enable the services that are absolutely necessary.

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*tftp:[\ \t](.*)$/ (Flags:Multiline,Comments mode)
tftp Service Status Excludes "on"

Remediation

To remediate failure of this policy test, disable the tftp service.

Disabling the tftp service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list tftp** command to check the status of the service.
3. Disable the **tftp** service using the **chkconfig tftp off** command.
4. Run the **/etc/init.d/xinetd restart** command to restart the **xinetd** service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="tftp"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig ${ServiceName} off 2>/dev/null
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null
 \
        | /bin/egrep -w "off"`
    if [ -z "$IsChanged" ]; then
        echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        echo "SUCCESS-3001: Startup mode of [$ServiceName]
 service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_RESTART_SERVICE xinetd
# AR_TEST_ID = T0000828
# AR_TEST_NAME = Verify That the tftp Service Is Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/sbin/service xinetd
 restart</b> command to restart the <b>xinetd </b>service.</li></
ol>
```

Post Remediation Category

```
Restart Service "xinetd"
```

Remediated Elements

```
/etc/xinetd.d/tftp
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/service xinetd restart** command to restart the **xinetd** service.

2.2.2. 9 Verify That the chargen-dgram Service Is Disabled

Verify That the chargen-dgram Service Is Disabled

Description	chargen-dram is a network service that responds with 0 to 512 ASCII characters for each datagram it receives. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*chargen-dgram:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode) chargen-dgram Excludes "on"
Remediation	To remediate failure of this policy test, disable the chargen-dgram service. **Disabling the chargen-dgram service**: 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list chargen-dgram** command to check the status of the service. 3. Disable the **chargen-dgram** service using the **chkconfig chargen-dgram off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2.10 Verify That the chargen-stream Service Is Disabled

Verify That the chargen-stream Service Is Disabled

Description	chargen-stream is a network service that responds with 0 to 512 ASCII characters for each connection it receives. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*chargen-stream:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode) chargen-stream Excludes "on"
Remediation	To remediate failure of this policy test, disable the chargen-stream service. **Disabling the chargen-stream service**: 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list chargen-stream** command to check the status of the service. 3. Disable the **chargen-stream** service using the **chkconfig chargen-stream off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2.11 Verify That the daytime-dgram Service Is Disabled

Verify That the daytime-dgram Service Is Disabled

Description	daytime-dram is a network service that responds with the server's current date and time. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*daytime-dgram:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode) daytime-dgram Excludes "on"
Remediation	To remediate failure of this policy test, disable the daytime-dgram service. **Disabling the daytime-dgram service**: 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list daytime-dgram** command to check the status of the service. 3. Disable the **daytime-dgram** service using the **chkconfig daytime-dgram off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2.12 Verify That the daytime-stream Service Is Disabled

Verify That the daytime-stream Service Is Disabled

Description	daytime-stream is a network service that responds with the server's current date and time. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*daytime-stream:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode) daytime-stream Excludes "on"
Remediation	To remediate failure of this policy test, disable the daytime-stream service. **Disabling the daytime-stream service**: 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list daytime-stream** command to check the status of the service. 3. Disable the **daytime-stream** service using the **chkconfig daytime-stream off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2.13 Verify That the echo-dgram Service Is Disabled

Verify That the echo-dgram Service Is Disabled

Description	echo-dgram is a network service that responds to clients with the data sent to it by the client. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*echo-dgram:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode) echo-dgram Excludes "on"
Remediation	To remediate failure of this policy test, disable the echo-dgram service. **Disabling the echo-dgram service**: 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list echo-dgram** command to check the status of the service. 3. Disable the **echo-dgram** service using the **chkconfig echo-dgram off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2.14 Verify That the echo-stream Service Is Disabled

Verify That the echo-stream Service Is Disabled

Description	echo-stream is a network service that responds to clients with the data sent to it by the client. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*echo-stream:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode) echo-stream Excludes "on"
Remediation	To remediate failure of this policy test, disable the echo-stream service. **Disabling the echo-stream service**: 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list echo-stream** command to check the status of the service. 3. Disable the **echo-stream** service using the **chkconfig echo-stream off** command. 4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service. For further details, please run the command **man chkconfig** to read man page.

2.2.2.15 Verify That the rexec Service Is Disabled

Verify That the rexec Service Is Disabled

Description

This test checks that the rexec service is disabled.
Because SSH provides a mechanism for both a secure login and a secure file transfer, it is not necessary for the insecure services that are supervised by xinetd to be running.
NOTE: It may be necessary for an organization to have one or more xinetd service running. If this is the case, only enable the services that are absolutely necessary.

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*rexec:[\ \t](.*)$/ (Flags:Multiline,Comments mode)
rexec Service Status Excludes "on"

Remediation

To remediate failure of this policy test, disable the rexec service.

Disabling the rexec service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list rexec** command to check status of the service.
3. Disable the **rexec** service using the **chkconfig rexec off** command.
4. Run the **/etc/init.d/xinetd restart** command to restart the **xinetd** service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="rexec"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig ${ServiceName} off 2>/dev/null
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null
 \
        | /bin/egrep -w "off"`
    if [ -z "$IsChanged" ]; then
        echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        echo "SUCCESS-3001: Startup mode of [$ServiceName]
 service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_RESTART_SERVICE xinetd
# AR_TEST_ID = T0004800
# AR_TEST_NAME = Verify That the rexec Service Is Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/sbin/service xinetd
 restart</b> command to restart the <b>xinetd </b>service.</li></
ol>
```

Post Remediation Category

```
Restart Service "xinetd"
```

Remediated Elements

```
/etc/xinetd.d/rexec
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/service xinetd restart** command to restart the **xinetd** service.

2.2.2.16 Verify That the rlogin Service Is Disabled

Verify That the rlogin Service Is Disabled

Description

This test checks that the rlogin service is disabled.
Because SSH provides a mechanism for both a secure login and a secure file transfer, it is not necessary for the insecure services that are supervised by xinetd to be running.
NOTE: It may be necessary for an organization to have one or more xinetd service running. If this is the case, only enable the services that are absolutely necessary.

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*rlogin:[\ \t](.*)$/ (Flags:Multiline,Comments mode)
rlogin Service Status Excludes "on"

Remediation

To remediate failure of this policy test, disable the rlogin service.

Disabling the rlogin service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list rlogin** command to check status of the service.
3. Disable the **rlogin** service using the **chkconfig rlogin off** command.
4. Run the **/etc/init.d/xinetd restart** command to restart the **xinetd** service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="rlogin"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig ${ServiceName} off 2>/dev/null
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null
 \
        | /bin/egrep -w "off"`
    if [ -z "$IsChanged" ]; then
        echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        echo "SUCCESS-3001: Startup mode of [$ServiceName]
 service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_RESTART_SERVICE xinetd
# AR_TEST_ID = T0004801
# AR_TEST_NAME = Verify That the rlogin Service Is Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/sbin/service xinetd
 restart</b> command to restart the <b>xinetd </b>service.</li></
ol>
```

Post Remediation Category

```
Restart Service "xinetd"
```

Remediated Elements

```
/etc/xinetd.d/rlogin
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/service xinetd restart** command to restart the **xinetd** service.

2.2.2.17 Verify That the rsh Service Is Disabled

Verify That the rsh Service Is Disabled

Description

This test checks that the rsh service is disabled.
Because SSH provides a mechanism for both a secure login and a secure file transfer, it is not necessary for the insecure services that are supervised by xinetd to be running.
NOTE: It may be necessary for an organization to have one or more xinetd service running. If this is the case, only enable the services that are absolutely necessary.

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*rsh:[\ \t](.*)$/ (Flags:Multiline,Comments mode)
rsh Service Status Excludes "on"

Remediation

To remediate failure of this policy test, disable the rsh service.

Disabling the rsh service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list rsh** command to check status of the service.
3. Disable the **rsh** service using the **chkconfig rsh off** command.
4. Run the **/etc/init.d/xinetd restart** command to restart the **xinetd** service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="rsh"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig ${ServiceName} off 2>/dev/null
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null \
        | /bin/egrep -w "off"`
    if [ -z "$IsChanged" ]; then
        echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        echo "SUCCESS-3001: Startup mode of [$ServiceName] service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_RESTART_SERVICE xinetd
# AR_TEST_ID = T0004802
# AR_TEST_NAME = Verify That the rsh Service Is Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an equivalent role.</li><li>Run the <b>/sbin/service xinetd restart</b> command to restart the <b>xinetd </b>service.</li></ol>
```

Post Remediation Category

```
Restart Service "xinetd"
```

Remediated Elements

```
/etc/xinetd.d/rsh
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/service xinetd restart** command to restart the **xinetd** service.

2.2.2.18 Verify That the talk Package Is Removed

Verify That the talk Package Is Removed

Description	The talk software makes it possible for users to send and receive messages across systems through a terminal session. The software presents a security risk as it uses unencrypted protocols for communication. It should be removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*talk-\d.*$/ (Flags:Multiline,Comments mode) talk Package Does not exist
Remediation	To remediate failure of this policy test, erase talk package. **Erasing talk package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase talk** command to remove **talk** package. For further details, please run the command **man yum** to read man page.

2.2.2.19 Verify That the talk-server Package Is Removed

Verify That the talk-server Package Is Removed

Description	The talk software makes it possible for users to send and receive messages across systems through a terminal session. The software presents a security risk as it uses unencrypted protocols for communication. It should be removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*talk-server-\d.*$/ (Flags:Multiline,Comments mode) talk-server Package Does not exist
Remediation	To remediate failure of this policy test, erase talk-server package. **Erasing talk-server package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase talk-server** command to remove **talk-server** package. For further details, please run the command **man yum** to read man page.

2.2.2.20 Verify That the avahi-daemon Service Is Disabled

Verify That the avahi-daemon Service Is Disabled

Description

This test checks that the avahi-daemon service is disabled.
All system daemons that do not have a clear and necessary purpose should be disabled. This greatly reduces the odds that a vulnerable system daemon will be targeted by an attack when an operating system vulnerability is discovered.

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*(?:avahi-daemon[\ \t]+.*on|Avahi[\ \t]+daemon[\ \t]+is[\ \t]+running|avahi-daemon[\ \t]+\(pid[\ \t]+\d+\)[\ \t]+is[\ \t]+running).*$/ (Flags:Multiline,Comments mode)
avahi-daemon Service Status Does not exist

Remediation

To remediate failure of this policy test, disable the avahi-daemon service.

Disabling the avahi-daemon service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list avahi-daemon** command to check status of the service.
3. Disable the **avahi-daemon** service using the **chkconfig --level 0123456 avahi-daemon off** command.
4. Run the **service avahi-daemon stop** command to stop the service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="avahi-daemon"
Level="0123456"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig --level ${Level} ${ServiceName} off 2>/dev/
null
    LevelRegex=`/bin/echo $Level | /bin/sed 's/\([0-6]\)/
[[:space:]]+\1:off/g'`
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null
 \
        | /bin/egrep "$LevelRegex"`
    if [ -z "$IsChanged" ]; then
        /bin/echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        /bin/echo "SUCCESS-3001: Startup mode of [$ServiceName]
 service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_BOOT_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_STOP_SERVICE avahi-daemon
# AR_TEST_ID = T0004891
# AR_TEST_NAME = Verify That the avahi-daemon Service Is Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>service avahi-daemon stop</
b> command to stop the <b>avahi-daemon </b>service.</li></ol>
```

Post Remediation Category

```
Stop Service "avahi-daemon"
```

Remediated Elements

```
/etc/rc.d/rc*.d/S*avahi-daemon
/etc/rc.d/rc*.d/K*avahi-daemon
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **service avahi-daemon stop** command to stop the **avahi-daemon** service.

2.2.2.21 Verify That the tcpmux-server Service Is Disabled

Verify That the tcpmux-server Service Is Disabled

Description

This test determines whether the tcpmux-server has been disabled. This setting supports system integrity and information confidentiality by preventing TCP port multiplexing (i.e. a rouge process using a well-known port to stay "under the radar").

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*tcpmux-server:[\ \t]+(.*)$/ (Flags:Multiline,Comments mode)
tcpmux-server Service Status Excludes "on"

Remediation

To remediate failure of this policy test, disable the tcpmux-server service.

Disabling the tcpmux-server service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list tcpmux-server** command to check the status of the service.
3. Disable the **tcpmux-server** service using the **chkconfig tcpmux-server off** command.
4. Run the **/sbin/service xinetd restart** command to restart **xinetd** service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="tcpmux-server"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig ${ServiceName} off 2>/dev/null
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null
 \
        | /bin/egrep -w "off"`
    if [ -z "$IsChanged" ]; then
        echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        echo "SUCCESS-3001: Startup mode of [$ServiceName]
 service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_RESTART_SERVICE xinetd
# AR_TEST_ID = T0013662
# AR_TEST_NAME = Verify That the tcpmux-server Service Is
 Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/sbin/service xinetd
 restart</b> command to restart the <b>xinetd </b>service.</li></
ol>
```

Post Remediation Category

```
Restart Service "xinetd"
```

Remediated Elements

```
/etc/xinetd.d/time
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/service xinetd restart** command to restart the **xinetd** service.

2.2.2.22 Verify That the xinetd Service Is Disabled

Verify That the xinetd Service Is Disabled

Description

This test checks that the xinetd service is disabled.
The xinetd service is used to supervise access to network services and should be disabled unless there is a compelling reason to be running one or more of the services that it supervises. Because SSH provides a mechanism for both a secure login and a secure file transfer, it is generally not necessary for the insecure services that are supervised by xinetd to be running.

Severity

0

Weight

5

Type

Content Test

Rules

Service Status

Element

Equals "service status"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*xinetd:?[\ \t](.*)$/ (Flags:Multiline,Comments mode)
xinetd Service Status Excludes "on"

Remediation

To remediate failure of this policy test, disable the xinetd service.

Disabling the xinetd service:

1. Become superuser or assume an equivalent role.
2. Run the **chkconfig --list xinetd** command to check the status of the service.
3. Disable the **xinetd** service using the **chkconfig --level 0123456 xinetd off** command.
4. Run the **service xinetd stop** command to stop the service.

For further details, please run the command **man chkconfig** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
ServiceName="xinetd"
Level="0123456"

# Issue the command to disable the service
IsExisted=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null`
if [ -n "$IsExisted" ]; then
    /sbin/chkconfig --level ${Level} ${ServiceName} off 2>/dev/
null
    LevelRegex=`/bin/echo $Level | /bin/sed 's/\([0-6]\)/
[[:space:]]+\1:off/g'`
    IsChanged=`/sbin/chkconfig --list ${ServiceName} 2>/dev/null
 \
        | /bin/egrep "$LevelRegex"`
    if [ -z "$IsChanged" ]; then
        /bin/echo "FAILURE-3001: Could not change startup mode"\
            "of [$ServiceName] service to disabled"
        exit 3001
    else
        /bin/echo "SUCCESS-3001: Startup mode of [$ServiceName]
 service"\
            "changed to disabled"
        exit 0
    fi
else
    echo "SUCCESS-3002: [$ServiceName] service does not exist"
    exit 0
fi

# AR_ACTION = RHEL_BOOT_SERVICE_DISABLE
# AR_COMPLETION = COMPLETION_STOP_SERVICE xinetd
# AR_TEST_ID = T0014810
# AR_TEST_NAME = Verify That the xinetd Service Is Disabled


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>service xinetd stop</b>
 command to stop the <b>xinetd </b>service.</li></ol>
```

Post Remediation Category

```
Stop Service "xinetd"
```

Remediated Elements

```
/etc/rc.d/rc*.d/S*xinetd
/etc/rc.d/rc*.d/K*xinetd
```

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **service xinetd stop** command to stop the **xinetd** service.

2.2.2.23 Verify That the rsh Package Is Removed

Verify That the rsh Package Is Removed

Description	The rsh package contains legacy services that exchange credentials in clear-text. It is recommended that The rsh package is removed.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*rsh-\d.*$/ (Flags:Multiline,Comments mode) rsh Package Does not exist
Remediation	To remediate failure of this policy test, erase rsh package. **Erasing rsh package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase rsh** command to remove **rsh** package. For further details, please run the command **man yum** to read man page.

2.2.2.24 Verify That the tftp Package Is Removed

Verify That the tftp Package Is Removed

Description	Trivial File Transfer Protocol (TFTP) is a simple file transfer protocol, typically used to automatically transfer configuration or boot files between machines. TFTP does not support authentication and can be easily hacked. The package tftp is a client program that allows for connections to a tftp server. It is recommended that TFTP be removed, unless there is a specific need for TFTP (such as a boot server).
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*tftp-\d.*$/ (Flags:Multiline,Comments mode) tftp Package Does not exist
Remediation	To remediate failure of this policy test, erase tftp package. **Erasing tftp package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase tftp** command to remove **tftp** package. For further details, please run the command **man yum** to read man page.

2.2.2.25 Verify That the tftp-server Package Is Removed

Verify That the tftp-server Package Is Removed

Description	Trivial File Transfer Protocol (TFTP) is a simple file transfer protocol, typically used to automatically transfer configuration or boot machines from a boot server. The package tftp-server is the server package used to define and support a TFTP server. It is recommended that TFTP be removed, unless there is a specific need for TFTP (such as a boot server).
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*tftp-server-\d.*$/ (Flags:Multiline,Comments mode) tftp-server Package Does not exist
Remediation	To remediate failure of this policy test, erase tftp-server package. **Erasing tftp-server package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase tftp-server** command to remove **tftp-server** package. For further details, please run the command **man yum** to read man page.

2.2.2.26 Verify That the xinetd Package Is Removed

Verify That the xinetd Package Is Removed

Description	The eXtended InterNET Daemon (xinetd) is an open source super daemon that replaced the original inetd daemon. The xinetd daemon listens for well known services and dis patches the appropriate daemon to properly respond to service requests. If there are no xinetd services required, it is recommended that the daemon be deleted from the system.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*xinetd-\d.*$/ (Flags:Multiline,Comments mode) xinetd Package Does not exist
Remediation	To remediate failure of this policy test, erase xinetd package. **Erasing xinetd package**: 1. Become superuser or assume an equivalent role. 2. Run the **yum erase xinetd** command to remove **xinetd** package. For further details, please run the command **man yum** to read man page.

2.2.4 System Security Configuration

Configure system security parameters to prevent misuse.

2.2.4. 1 Verify That Users Are Assigned Valid Home Directories

Verify That Users Are Assigned Valid Home Directories

<table>
<tr><th></th><th></th></tr>
<tr><td>**Description**</td><td>The /etc/passwd file defines a home directory that the user is placed in upon login. If the user's home directory does not exist or is unassigned, the user will be placed in "/" and will not be able to write any files or have local environment variables set.</td></tr>
<tr><td>**Severity**</td><td>0</td></tr>
<tr><td>**Weight**</td><td>5</td></tr>
<tr><td>**Type**</td><td>Content Test</td></tr>
<tr><td>**Rules**</td><td>User Home Directories</td></tr>
<tr><td>**Element**</td><td>Equals "User Home Directories"</td></tr>
<tr><td>**Version conditions**</td><td>If an element version has no content, the condition should:Pass
Regular expression: /^[\\t]*UserName=(?!nfsnobody[\ \t])\S+[\ \t]+UserID=(?:[5-9]\d{2}|0*\d{4,})[\ \t]+(?:UserHome=[\ \t]+Permissions.*|.*HomeDirExisted=no)$/ (Flags:Multiline,Comments mode)
User That Not Be Assigned Valid Home Directories Does not exist</td></tr>
<tr><td>**Remediation**</td><td>To remediate failure of this policy test, assign valid home directory for all normal users.
Assigning valid home directory for all normal users:
1. Become superuser or assume an equivalent role.
2. Run the following command to list all user that not be a valid home directory: **Users=`/bin/cat /etc/passwd 2>/dev/null | /bin/egrep -v "^[[:space:]]*(#.*|\+.*|nfsnobody):" | /bin/awk -F: '$3 >= 500 {print}'`; SavedIFS="$IFS"; IFS=`/bin/echo -en "\n\b"`; for User in $Users; do UserName=`echo $User | /bin/awk -F: '{print $1}'`;UserHome=`/bin/echo $User | /bin/awk -F: '{print $6}'`; if ["$UserHome" != "/"]; then if [! -d "$UserHome"]; then /bin/echo $UserName ; fi; fi; done; IFS="$SavedIFS"**
3. Run the **usermod -d <home_directory> <user_name>** command to assign a home directory for users found in the step 2.</td></tr>
</table>

2.2.4. 2 Verify That .forward Files Are Not Used

Verify That .forward Files Are Not Used

Description	This test verifies that .forward files are not used. An attacker that gains access to a .forward file can turn the host into a spam producing system or hijack user email.
Severity	0
Weight	5
Type	Content Test
Rules	User Dot Files
Element	Equals "User Dot Files"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^.*/\.forward$/ (Flags:Multiline,Comments mode) .forward File Does not exist
Remediation	To remediate failure of this policy test, remove the .forward files in the user home directories.

Removing the .forward files in the user home directories:

1. Become superuser or assume an equivalent role.
2. Run the script:

```
Users=`/bin/egrep -v "^[[:space:]]*#|^[[:space:]]*$" /etc/pass
wd 2>/dev/null | /bin/awk -F: '{ cmd = "/usr/bin/passwd -S "
$1 " 2>/dev/null"; cmd | getline UserInfo; if ($0 !~ /^[[:space:
]]*(#.*|\+.*|root|halt|sync|shutdown):/ && (UserInfo ~ /^[[:g
raph:]]+[[:space:]]+PS[[:space:]]+/ || (UserInfo ~ /^[[:space:
]]*Unknown[[:space:]]+user\./ && $2 != "!!")) && $7 !~ /^\/sbin
\/nologin$/){ print $1 ":" $6}}'`; SavedIFS="$IFS";  IFS=`/bin/
echo -e "\n\b"`; for User in $Users; do UserName=`/bin/echo
"$User" | /bin/awk -F: '{print $1}'`; HomeDirectory=`/bin/echo
"$User" | /bin/awk -F: '{print $2}'`; /bin/ls -alL $HomeDirec
tory/.forward 2>/dev/null | awk '$1 !~ /^d/ { FileName=subst
r($0,index($0,"/")); print UserName, $1, $3, $4, FileName}'
UserName="$UserName"; done; IFS="$SavedIFS";
```

   to list all **.forward** files.
3. Remove **.forward** files found in step 2 using the **rm -f <.forward_file_name>** command.

For further details, please run the command **man rm** to read man page.

2.2.4. 3 Verify That .netrc Files Do Not Exist

Verify That .netrc Files Do Not Exist

Description

This test determines if any .netrc files are present on the system. These files may contain unencrypted passwords which could be used to attack other systems. Examine the list of files found by this policy test very carefully and identify application dependencies and user impact before removing anything.

Severity

0

Weight

5

Type

Content Test

Rules

User Dot Files

Element

Equals "User Dot Files"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^.*/\.netrc$/ (Flags:Multiline,Comments mode)
.netrc File Does not exist

Remediation

To remediate failure of this policy test, remove the .netrc files in the user home directories.

Removing the .netrc files in the user home directories:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   Users=`(/bin/egrep -v "^[[:space:]]*#|^[[:space:]]*$" /etc/passwd 2>/dev/null) | /bin/awk -F: '{ cmd = "/usr/bin/passwd -S " $1 " 2>/dev/null"; cmd | getline UserInfo; if ($0 !~ /^[[:space:]]*(#.*|\+.*|root|halt|sync|shutdown):/ && (UserInfo ~ /^[[:graph:]]+[[:space:]]+PS[[:space:]]+/ || (UserInfo ~ /^[[:space:]]*Unknown[[:space:]]+user\./ && $2 != "!!")) && $7 !~ /^\/sbin\/nologin$/){ print $1 ":" $6}}'`; SavedIFS="$IFS"; IFS=`/bin/echo -e "\n\b"`; for User in $Users; do UserName=`/bin/echo "$User" | /bin/awk -F: '{print $1}'`; HomeDirectory=`/bin/echo "$User" | /bin/awk -F: '{print $2}'`; /bin/ls -alL $HomeDirectory/.netrc 2>/dev/null | awk '$1 !~ /^d/ { FileName=substr($0, index($0,"/")); print UserName, $1, $3, $4, FileName}' UserName="$UserName"; done; IFS="$SavedIFS";
   ```

   to list all **.netrc** files.
3. Remove **.netrc** files found in step 2 using the **rm -f <.netrc_file_name>** command.

For further details, please run the command **man rm** to read man page.

2.2.4. 4 Verify That the Screensaver Is Configured to Blank

Verify That the Screensaver Is Configured to Blank

Description

Setting the screensaver mode to blank-only conceals the contents of the display from passersby.

Severity

0

Weight

5

Type

Content Test

Rules

Get Gnome Screensaver Mode

Element

Equals "Gnome Screensaver Mode"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*blank-only[\ \t]*$/ (Flags:Multiline,Comments mode)
Blank-only Mode Exists

Remediation

To remediate failure of this policy test, set the screensaver mode in the Gnome desktop to a blank screen.

Setting the screensaver mode in the Gnome desktop to a blank screen:

1. Become superuser or assume an equivalent role.
2. Install **GConf2** package if it is not installed:

```
/bin/rpm -ivh <GConf2 package>
```

3. Run the following command to set the screensaver mode in the GNOME desktop to a blank screen:

```
gconftool-2 --direct --config-source xml:readwrite:/etc/gconf/gconf.xml.mandatory --type string
--set /apps/gnome-screensaver/mode blank-only
```

For further details, please refer to:

http://projects.gnome.org/gconf/

2.2.4. 5 Verify That the System Locks Accounts after Three or Less Consecutive Unsuccessful Login Attempts (password-auth)

Verify That the System Locks Accounts after Three or Less Consecutive Unsuccessful Login Attempts (password-auth)

Description

This test verifies that the system locks accounts after three or less consecutive unsuccessful login attempts.
Locking out user accounts after a number of incorrect attempts prevents direct password guessing attacks.

Severity

0

Weight

5

Type

Content Test

Rules

Get auth Modules Configured in /etc/pam.d/password-auth File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "auth Configuration"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bsuccess=1\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authsucc\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$|^[\ \t]*auth[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+preauth\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|requisite|required|\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$\n(?:^.*$\n)*?^[\ \t]*account[\ \t]+[^\#&&\S]*[\ \t]+[^\#&&\S]*\bpam_faillock\.so\b.*$/ (Flags:Multiline,Comments mode)
Faillock Module Setting Exists AND
((Number of authfail Unsuccessful Login Attempts Is between 1 AND 3 AND
Number of authsucc Unsuccessful Login Attempts Is between 1 AND 3) OR
(Number of preauth Unsuccessful Login Attempts Is between 1 AND 3 AND
Number of authfail Unsuccessful Login Attempts Is between 1 AND 3) OR
(Number of authfail Unsuccessful Login Attempts Does not exist AND
Number of authsucc Unsuccessful Login Attempts Does not exist AND
Number of preauth Unsuccessful Login Attempts Does not exist AND
Number of authfail Unsuccessful Login Attempts Does not exist))

Remediation

To remediate failure of this policy test, configure the system to lock accounts after three or less consecutive unsuccessful login attempts (password-auth).

Configuring the system to lock accounts after three or less consecutive unsuccessful login attempts (password-auth):

1. Become superuser or assume an equivalent role.
2. Edit the **/etc/pam.d/password-auth** file, update or add the lines below to the file:

   auth [success=1] pam_unix.so
   auth [default=die] pam_faillock.so authfail deny=*<value>* *[other parameters]*
   auth sufficient pam_faillock.so authsucc deny=*<value>* *[other parameters]*

   Or:

   auth [requisite|required] pam_faillock.so preauth deny=*<value>* *[other parameters]*
   auth sufficient pam_unix.so
   auth [sufficient|requisite|required] pam_faillock.so authfail deny=*<value>* *[other parameters]*
   [other_rules]
   account required pam_faillock.so

   where ***<value>*** is equal to **1**, **2** or **3**.
3. Save the file.

Note: The ***<value>*** should be a same value on each line.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

2.2.4. 6 Verify That the Unlock Time of an Account Is Less than or Equal to 604800, but Not 0 Seconds (password-auth)

Verify That the Unlock Time of an Account Is Less than or Equal to 604800, but Not 0 Seconds (password-auth)

Description

Locking out user accounts after a number of incorrect attempts prevents direct password guessing attacks. Ensuring that an administrator is involved in unlocking locked accounts draws appropriate attention to such situations.

Severity

0

Weight

5

Type

Content Test

Rules

Get auth Modules Configured in /etc/pam.d/password-auth File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "auth Configuration"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bsuccess=1\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authsucc\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$|^[\ \t]*auth[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+preauth\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|requisite|required|\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$\n(?:^.*$\n)*?^[\ \t]*account[\ \t]+[^\#&&\S]*[\ \t]+[^\#&&\S]*\bpam_faillock\.so\b.*$/ (Flags:Multiline,Comments mode)
Faillock Module Setting Exists AND
((authfail Unlock Time of an Account Is between 1 AND 604800 AND
authsucc Unlock Time of an Account Is between 1 AND 604800) OR
(preauth Unlock Time of an Account Is between 1 AND 604800 AND
authfail Unlock Time of an Account Is between 1 AND 604800) OR
(authfail Unlock Time of an Account Does not exist AND
authsucc Unlock Time of an Account Does not exist AND
preauth Unlock Time of an Account Does not exist AND
authfail Unlock Time of an Account Does not exist))

Remediation

To remediate failure of this policy test, configure the authentication system to unlock an account due to excessive failed login attempts after one week or less, but not 0 (password-auth).

Configuring the authentication system to unlock an account due to excessive failed login attempts after one week or less, but not 0 (password-auth):

1. Become superuser or assume an equivalent role.
2. Edit the **/etc/pam.d/password-auth** file, update or add the lines below to the file:

   auth [success=1] pam_unix.so
   auth [default=die] pam_faillock.so authfail unlock_time=<*value*> *[other parameters]*
   auth sufficient pam_faillock.so authsucc unlock_time=<*value*> *[other parameters]*

   Or:

   auth [requisite|required] pam_faillock.so preauth unlock_time=<*value*> *[other parameters]*
   auth sufficient pam_unix.so
   auth [sufficient|requisite|required] pam_faillock.so authfail unlock_time=<*value*> *[other parameters]*
   [other_rules]
   account required pam_faillock.so

   where **<*value*>** is equal to **604800** or **less**, but not **0**.
3. Save the file.

Note: The **<*value*>** should be a same value on each line.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

2.2.4. 7 Verify That the System Is Configured to Disable Accounts after Excessive Login Failures within a 15-minute Interval (password-auth)

Verify That the System Is Configured to Disable Accounts after Excessive Login Failures within a 15-minute Interval (password-auth)

Description	This test verifies that the system must disable accounts after excessive login failures within a 15-minute interval. Locking out user accounts after a number of incorrect attempts within a specific period of time prevents direct password guessing attacks.
Severity	0
Weight	5
Type	Content Test
Rules	Get auth Modules Configured in /etc/pam.d/password-auth File
Excluded Nodes	Red Hat Enterprise Linux Server 6
Element	Equals "auth Configuration"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bsuccess=1\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bfail_interval=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authsucc\b(?:[\ \t]+[^\#\n]*\bfail_interval=(\d+)\b)?.*$\|^[\ \t]*auth[\ \t]+(?:requisite\|required)[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+preauth\b(?:[\ \t]+[^\#\n]*\bfail_interval=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+(?:sufficient\|requisite\|required\|\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bfail_interval=(\d+)\b)?.*$\n(?:^.*$\n)*?^[\ \t]*account[\ \t]+[^\#&&\S]*[\ \t]+[^\#&&\S]*\bpam_faillock\.so\b.*$/ (Flags:Multiline,Comments mode) Faillock Module Setting Exists AND ((authfail Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900 AND authsucc Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900) OR (preauth Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900 AND authfail Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900) OR (authfail Interval of Disabling Accounts after Excessive Login Failures Does not exist AND authsucc Interval of Disabling Accounts after Excessive Login Failures Does not exist AND preauth Interval of Disabling Accounts after Excessive Login Failures Does not exist AND authfail Interval of Disabling Accounts after Excessive Login Failures Does not exist))
Remediation	To remediate failure of this policy test, set the length of the interval during which the consecutive authentication failures must happen for the user account lock out to 900 seconds or greater (password-auth). **Setting the length of the interval during which the consecutive authentication failures must happen for the user account lock out to 900 seconds or greater (password-auth):** 1. Become superuser or assume an equivalent role. 2. Edit the **/etc/pam.d/password-auth** file, update or add the lines below to the file: **auth [success=1] pam_unix.so** **auth [default=die] pam_faillock.so authfail fail_interval=<*value*> *[other parameters]*** **auth sufficient pam_faillock.so authsucc fail_interval=<*value*> *[other parameters]*** Or: **auth [requisite|required] pam_faillock.so preauth fail_interval=<*value*> *[other parameters]*** **auth sufficient pam_unix.so** **auth [sufficient|requisite|required] pam_faillock.so authfail fail_interval=<*value*> *[other parameters]*** ***[other_rules]*** **account required pam_faillock.so** where **<*value*>** is equal to **900** or **greater**. 3. Save the file. **Note**: The **<*value*>** should be a same value on each line. For further details, please refer to: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

2.2.4. 8 Verify That the System Disables Accounts after Three or Less Consecutive Unsuccessful Login Attempts (system-auth)

Verify That the System Disables Accounts after Three or Less Consecutive Unsuccessful Login Attempts (system-auth)

Description

This test verifies that the system locks accounts after three or less consecutive unsuccessful login attempts.
Locking out user accounts after a number of incorrect attempts prevents direct password guessing attacks.

Severity

0

Weight

5

Type

Content Test

Rules

Get All Configurations in system-auth File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Get All Configurations in system-auth File"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bsuccess=1\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authsucc\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$|^[\ \t]*auth[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+preauth\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|requisite|required|\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bdeny=(\d+)\b)?.*$\n(?:^.*$\n)*?^[\ \t]*account[\ \t]+[^\#&&\S]*[\ \t]+[^\#&&\S]*\bpam_faillock\.so\b.*$/ (Flags:Multiline,Comments mode)
Faillock Module Setting Exists AND
((Number of authfail Unsuccessful Login Attempts Is between 1 AND 3 AND
Number of authsucc Unsuccessful Login Attempts Is between 1 AND 3) OR
(Number of preauth Unsuccessful Login Attempts Is between 1 AND 3 AND
Number of authfail Unsuccessful Login Attempts Is between 1 AND 3) OR
(Number of authfail Unsuccessful Login Attempts Does not exist AND
Number of authsucc Unsuccessful Login Attempts Does not exist AND
Number of preauth Unsuccessful Login Attempts Does not exist AND
Number of authfail Unsuccessful Login Attempts Does not exist))

Remediation

To remediate failure of this policy test, configure the system to lock accounts after three or less consecutive unsuccessful login attempts (system-auth).

Configuring the system to lock accounts after three or less consecutive unsuccessful login attempts (system-auth):

1. Become superuser or assume an equivalent role.
2. Edit the **/etc/pam.d/system-auth** file, update or add the lines below to the file:

   auth [success=1] pam_unix.so
   auth [default=die] pam_faillock.so authfail deny=<*value*> ***[other parameters]***
   auth sufficient pam_faillock.so authsucc deny=<*value*> ***[other parameters]***

   Or:

   auth [requisite|required] pam_faillock.so preauth deny=<*value*> ***[other parameters]***
   auth sufficient pam_unix.so
   auth [sufficient|requisite|required] pam_faillock.so authfail deny=<*value*> ***[other parameters]***
   [other_rules]
   account required pam_faillock.so

   where **<*value*>** is equal to **1**, **2** or **3**.
3. Save the file.

Note: The **<*value*>** should be a same value on each line.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

2.2.4. 9 Verify That the Unlock Time of an Account Is Less than or Equal to 604800, but Not 0 Seconds (system-auth)

Verify That the Unlock Time of an Account Is Less than or Equal to 604800, but Not 0 Seconds (system-auth)

Description

Locking out user accounts after a number of incorrect attempts prevents direct password guessing attacks. Ensuring that an administrator is involved in unlocking locked accounts draws appropriate attention to such situations.

Severity

0

Weight

5

Type

Content Test

Rules

Get All Configurations in system-auth File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Get All Configurations in system-auth File"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bsuccess=1\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authsucc\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$|^[\ \t]*auth[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+preauth\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|requisite|required|\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bunlock_time=(\d+)\b)?.*$\n(?:^.*$\n)*?^[\ \t]*account[\ \t]+[^\#&&\S]*[\ \t]+[^\#&&\S]*\bpam_faillock\.so\b.*$/ (Flags:Multiline,Comments mode)
Faillock Module Setting Exists AND
((authfail Unlock Time of an Account Is between 1 AND 604800 AND
authsucc Unlock Time of an Account Is between 1 AND 604800) OR
(preauth Unlock Time of an Account Is between 1 AND 604800 AND
authfail Unlock Time of an Account Is between 1 AND 604800) OR
(authfail Unlock Time of an Account Does not exist AND
authsucc Unlock Time of an Account Does not exist AND
preauth Unlock Time of an Account Does not exist AND
authfail Unlock Time of an Account Does not exist))

Remediation

To remediate failure of this policy test, configure the authentication system to unlock an account due to excessive failed login attempts after one week or less, but not 0 (system-auth).

Configuring the authentication system to unlock an account due to excessive failed login attempts after one week or less, but not 0 (system-auth):

1. Become superuser or assume an equivalent role.
2. Edit the **/etc/pam.d/system-auth** file, update or add the lines below to the file:

   auth [success=1] pam_unix.so
   auth [default=die] pam_faillock.so authfail unlock_time=<*value*> *[other parameters]*
   auth sufficient pam_faillock.so authsucc unlock_time=<*value*> *[other parameters]*

   Or:

   auth [requisite|required] pam_faillock.so preauth unlock_time=<*value*> *[other parameters]*
   auth sufficient pam_unix.so
   auth [sufficient|requisite|required] pam_faillock.so authfail unlock_time=<*value*> *[other parameters]*
   [other_rules]
   account required pam_faillock.so

   where **<*value*>** is equal to **604800** or **less**, but not **0**.
3. Save the file.

Note: The **<*value*>** should be a same value on each line.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

2.2.4.10 Verify That the System Is Configured to Disable Accounts after Excessive Login Failures within a 15-minute Interval (system-auth)

Verify That the System Is Configured to Disable Accounts after Excessive Login Failures within a 15-minute Interval (system-auth)

Description

This test verifies that the system must disable accounts after excessive login failures within a 15-minute interval. Locking out user accounts after a number of incorrect attempts within a specific period of time prevents direct password guessing attacks.

Severity

0

Weight

5

Type

Content Test

Rules

Get All Configurations in system-auth File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Get All Configurations in system-auth File"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bsuccess=1\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*
\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+\[[^\#\]\n]*\bdefault=die\b[^\#\]\n]*\][\ \t]+[^\#&&\S]*
\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#\n]*\bfail_interval=(\d+)\b)?.*$\n^[\ \t]*auth[\
\t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[
\ \t]+authsucc\b(?:[\ \t]+[^\#\n]*\bfail_interval=(\d+)\b)?.*$|^[\ \t]*auth[\ \t]+(?:requisite|re
quired)[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+preauth\b(?:[\ \t]+[^\#\n]*\bfail_interval
=(\d+)\b)?.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|\[[^\#\]\n]*\bsuccess=done\b[^\#\]\n]*\])[\ \t]+[
^\#&&\S]*\bpam_unix\.so\b.*$\n^[\ \t]*auth[\ \t]+(?:sufficient|requisite|required|\[[^\#\]\n]
\bdefault=die\b[^\#\]\n]\])[\ \t]+[^\#&&\S]*\bpam_faillock\.so[\ \t]+authfail\b(?:[\ \t]+[^\#
\n]*\bfail_interval=(\d+)\b)?.*$\n(?:^.*$\n)*?^[\ \t]*account[\ \t]+[^\#&&\S]*[\ \t]+[^\#&&\S]*
\bpam_faillock\.so\b.*$/ (Flags:Multiline,Comments mode)
Faillock Module Setting Exists AND
((authfail Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900 AND
authsucc Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900) OR
(preauth Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900 AND
authfail Interval of Disabling Accounts after Excessive Login Failures Greater than or equal 900) OR
(authfail Interval of Disabling Accounts after Excessive Login Failures Does not exist AND
authsucc Interval of Disabling Accounts after Excessive Login Failures Does not exist AND
preauth Interval of Disabling Accounts after Excessive Login Failures Does not exist AND
authfail Interval of Disabling Accounts after Excessive Login Failures Does not exist))

Remediation

To remediate failure of this policy test, set the length of the interval during which the consecutive authentication failures must happen for the user account lock out to 900 seconds or greater (system-auth).

Setting the length of the interval during which the consecutive authentication failures must happen for the user account lock out to 900 seconds or greater (system-auth):

1. Become superuser or assume an equivalent role.
2. Edit the **/etc/pam.d/system-auth** file, update or add the lines below to the file:

   auth [success=1] pam_unix.so
   auth [default=die] pam_faillock.so authfail fail_interval=<*value*> *[other parameters]*
   auth sufficient pam_faillock.so authsucc fail_interval=<*value*> *[other parameters]*

   Or:

   auth [requisite|required] pam_faillock.so preauth fail_interval=<*value*> *[other parameters]*
   auth sufficient pam_unix.so
   auth [sufficient|requisite|required] pam_faillock.so authfail fail_interval=<*value*> *[other parameters]*
   [other_rules]
   account required pam_faillock.so

   where **<*value*>** is equal to **900** or **greater**.
3. Save the file.

Note: The **<*value*>** should be a same value on each line.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

2.2.4.11 Verify the idle_delay Is Less than or Equal to 15

Verify the idle_delay Is Less than or Equal to 15

Description

This test verifies idle_delay is less than or equal to 15.
If graphical desktop sessions do not lock the session after 15 minutes of inactivity, requiring re-authentication to resume operations, the system or individual data could be compromised by an alert intruder who could exploit the oversight. This requirement applies to graphical desktop environments provided by the system to locally attached displays and input devices as well as to graphical desktop environments provided to remote systems, including thin clients.

Severity

0

Weight

5

Type

Content Test

Rules

Get the idle_delay Value of Gnome Screensaver

Element

Equals "Gnome Screen Saver"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode)
Idle Timeout Less than or equal 15 AND
Idle Timeout Greater than 0

Remediation

To remediate failure of this policy test, set the idle_delay to less than or equal to 15.

Setting the idle_delay to less than or equal to 15:

1. Become superuser or assume an equivalent role.
2. Install **GConf2** package if it is not installed:

   /bin/rpm -ivh <GConf2 package>
3. Run the following command to set the idle timeout to less than or equal to 15:

   ```
   gconftool-2 --direct --config-source xml:readwrite:/etc/gconf/gconf.xml.mandatory --type int --set /desktop/gnome/session/idle_delay <value>
   ```
   where **<value>** is less than or equal to **15**.

For further details, please refer to:

http://projects.gnome.org/gconf/

2.2.4.12 Verify the Default maxlogins

Verify the Default maxlogins

Description

This test verifies that default maxlogins is set to a value less than or equal 10 but greater than 0.
Limiting simultaneous user logins can insulate the system from denial of service problems caused by excessive logins. Automated login processes operating improperly or maliciously may result in an exceptional number of simultaneous login sessions.

Severity

0

Weight

5

Type

Content Test

Rules

Get Content of Configuration Files for the pam_limits Module

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Contains "Get Content of Configuration Files for the pam_limits Module"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]**[\ \t]+hard+[\ \t]+maxlogins[\ \t]+(\d+)[\ \t]*(?:$|\#)/ (Flags:Multiline,Case insensitive,Comments mode)
maxlogins Parameter Less than or equal 10 AND
maxlogins Parameter Greater than 0

Remediation

To remediate failure of this policy test, set default maxlogins to a value less than or equal 10 and greater than 0.

Setting default maxlogins to a value less than or equal 10 and greater than 0:

1. Become superuser or assume an equivalent role.
2. Run following command to get the default maxlogins setting:

   ```
   /bin/egrep 'hard[[:space:]]+maxlogins' /etc/security/limits.conf /etc/security/limits.d/*.conf 2>/dev/null
   ```
3. If step 2 returns a file with maxlogins setting, open file and edit line such as: * **hard maxlogins <value>**
   <value>
4. If step 2 does not return anything, open the **/etc/security/limits.conf** file and add the line such as: * **hard maxlogins <value>**
   <value>

For further details, please run the command **man limits.conf** to read man page.

2.2.4.13 Verify the /etc/passwd File Does Not Contain Password Hashes

Verify the /etc/passwd File Does Not Contain Password Hashes

Description

This test verifies the /etc/passwd file does not contain password hashes.
If password hashes are readable by non-administrators, the passwords are subject to attack through lookup tables or cryptographic weaknesses in the hashes.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/passwd"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[^\#:]+:(?!(x|*|!|!!)?:).*$
/ (Flags:Multiline,Comments mode)
Password Hash Does not exist

Remediation

To remediate failure of this policy test, migrate **/etc/passwd** password hashes to **/etc/shadow** file.

Migrate /etc/passwd password hashes to /etc/shadow:

1. Become superuser or assume an equivalent role.
2. Migrate **/etc/passwd** password hashes to **/etc/shadow** file, using this command **/usr/sbin/pwconv**

For further details, please refer to:

https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/4/html/Introduction_To_System_Administration/s2-acctsgrps-files.html

2.2.4.14 Verify That A File Integrity Tool Is Used At Least Weekly

Verify That A File Integrity Tool Is Used At Least Weekly

Description	This test verifies that a file integrity tool must be used at least weekly. Changes in system libraries, binaries and other critical system files can indicate compromise or significant system events such as patching needing to be checked by automated processes and the results reviewed by the SA.
Severity	0
Weight	5
Type	Content Test
Rules	Verify That A File Integrity Tool Is Used At Least Weekly
Excluded Nodes	Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/crontab, /etc/cron.d/*, /etc/cron.weekly/*"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*AIDE[\ \t]+is[\ \t]+configured[\ \t]+to[\ \t]+run[\ \t]+weekly[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) AIDE Config Run Weekly Exists
Remediation	To remediate failure of this policy test, establish an automated job, scheduled to run weekly or more frequently. **Establishing an automated job, scheduled to run weekly or more frequently:** 1. Become superuser or assume an equivalent role. 2. Run the command **yum list aide** to check if aide is intalled or not. 3. If aide is not installed, go to the directory that contains aide package, run com mand **yum install <aide_package**> to install aide tool 4. Execute the following command: **crontab -u root -e** 5. Add the line to run aide weekly, example: **0 0 * * 0 /usr/sbin/aide --check** 6. Save files to apply the change. Note: This configuration run aide once a week at midnight in the morning of Sunday. For further details, please run the command **man aide** to read man page.

2.2.4.15 Verify That USB Mass Storage Is Prevented from Dynamic Loading

Verify That USB Mass Storage Is Prevented from Dynamic Loading

Description	This test verifies that the USB Mass Storage is prevented from dynamic loading. USB is a common computer peripheral interface. USB devices may include storage devices with the potential to install malicious software on a system or exfiltrate data.
Severity	0
Weight	5
Type	Content Test
Rules	Check modprobe Config
Element	Equals "modprobe Config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^\#\n]*\binstall[\ \t]+usb-storage[\ \t]+\S+.*$/ (Flags:Multiline,Comments mode) USB Mass Storage Disabling Exists
Remediation	To remediate failure of this policy test, prevent USB Mass Storage protocol handler from dynamic loading **To prevent USB Mass Storage protocol handler from dynamic loading:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/modprobe.conf** file or create a new configuration file under the **/etc/modprobe.d** directory. 3. Add the **install usb-storage /bin/true** entry to the file. 4. Save the file. For further details, please run the command **man modprobe.conf** to read man page.

2.2.4.16 Verify That Console Screen Locking Is Enabled

Verify That Console Screen Locking Is Enabled

Description	This test verifies that the system allows locking of the console screen. Installing "screen" ensures a console locking capability is available for users who may need to suspend console logins.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^screen.*$/ (Flags:Multiline,Comments mode) Console Screen Locking Exists
Remediation	To remediate failure of this policy test, install the screen package. **Installing the screen package**: 1. Become superuser or assume an equivalent role. 2. Run the **rpm -ivh <rpm_screen_package>** command to install the **screen** package.

2.2.4.17 Verify That the Gnome Desktop Keybinding for Locking the Screen Is Set

Verify That the Gnome Desktop Keybinding for Locking the Screen Is Set

Description

This test verifies that the system allows locking of graphical desktop sessions. The ability to lock graphical desktop sessions manually allows users to easily secure their accounts should they need to depart from their workstations temporarily.

Severity

0

Weight

5

Type

Content Test

Rules

Get the Keybindings in the Gnome Screensaver

Element

Equals "Get the Keybindings in the Gnome Screensaver"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /.+/ (Flags:Case insensitive)
Gnome Package Is Not Install or Gnome Desktop Keybinding Setting Exists

Remediation

To remediate failure of this policy test, set the desktop keybinding in the Gnome screen saver.

Setting the desktop keybinding in the Gnome screensaver:

1. Become superuser or assume an equivalent role.
2. Install **GConf2** package if it is not installed:

```
/bin/rpm -ivh <GConf2 package>
```

3. Run the following command to set the desktop keybinding in the Gnome screen saver:

```
/usr/bin/gconftool-2 --direct --config-source

xml:readwrite:/etc/gconf/gconf.xml.mandatory --
type string --set /apps/gnome_settings_daemo
n/keybindings/screensaver "<key_sequence>"
```

   where **"<key_sequence>"** is the keybinding for locking the screen (**<Control><Alt>l** is the default).

For further details, please refer to:

http://projects.gnome.org/gconf/

2.2.4.18 Verify That a /tmp Partition Is in the /etc/fstab File

Verify That a /tmp Partition Is in the /etc/fstab File

Description	The /tmp directory is a world-writable directory used for temporary storage by all users and some applications. Creating a separate partition for /tmp avoids a risk of resource exhaustion.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^\#\n]*[\ \t]+/tmp[\ \t]+.*$/ (Flags:Multiline,Comments mode) /tmp Entry Exists
Remediation	To remediate failure of this policy test, create separate a partition for /tmp. **Creating a separate partition for /tmp**: 1. For new installations, check the box to "**Review and modify partitioning**" and create a separate partition for **/tmp**. 2. For systems that were previously installed, use the **Logical Volume Manager (LVM)** to create partitions. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

2.2.4.19 Verify That a /var Partition Is in the /etc/fstab File

Verify That a /var Partition Is in the /etc/fstab File

Description	The /var directory is used by daemons and other system services to temporarily store dynamic data. Some directories created by these processes may be world-writable. Creating a separate partition for /var avoids a risk of resource exhaustion.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^\#\n]*[\ \t]+/var[\ \t]+.*$/ (Flags:Multiline,Comments mode) /var Entry Exists
Remediation	To remediate failure of this policy test, create a separate partition for /var. **Creating a separate partition for /var**: 1. For new installations, check the box to "**Review and modify partitioning**" and create a separate partition for **/var**. 2. For systems that were previously installed, use the **Logical Volume Manager (LVM)** to create partitions. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

2.2.4.20 Verify That a /var/log Partition Is in the /etc/fstab File

Verify That a /var/log Partition Is in the /etc/fstab File

Description	The /var/log directory is used by system services to store log data. There are two important reasons to ensure that system logs are stored on a separate partition: protection against resource exhaustion (since logs can grow quite large) and protection of audit data.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^\#\n]*[\ \t]+/var/log[\ \t]+.*$/ (Flags:Multiline,Comments mode) /var/log Entry Exists
Remediation	To remediate failure of this policy test, create a separate partition for /var/log. **Creating a separate partition for /var/log**: 1. For new installations, check the box to "**Review and modify partitioning**" and create a separate partition for **/var/log**. 2. For systems that were previously installed, use the **Logical Volume Manager (LVM)** to create partitions. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

2.2.4.21 Verify That a /var/log/audit Partition Is in the /etc/fstab File

Verify That a /var/log/audit Partition Is in the /etc/fstab File

Description	The /var/log/audit directory is used to store log data created the auditing daemon, auditd. There are two important reasons to ensure that system logs are stored on a separate partition: protection against resource exhaustion (since logs can grow quite large) and protection of audit data.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^\#\n]*[\ \t]+/var/log/audit[\ \t]+.*$/ (Flags:Multiline,Comments mode) /var/log/audit Entry Exists
Remediation	To remediate failure of this policy test, create separate a partition for /var/log/audit. **Creating a separate partition for /var/log/audit**: 1. For new installations, check the box to "**Review and modify partitioning**" and create a separate partition for **/var/log/audit**. 2. For systems that were previously installed, use the **Logical Volume Manager (LVM)** to create partitions. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

2.2.4.22 Verify That a /home Partition Is in the /etc/fstab File

Verify That a /home Partition Is in the /etc/fstab File

Description	The /home directory is used to support disk storage needs of local users. If the system is intended to support local users, create a separate partition for the /home directory to protect against resource exhaustion and restrict the type of files that can be stored under /home.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[^\#\n]*[\ \t]+/home[\ \t]+.*$/ (Flags:Multiline,Comments mode) /home Entry Exists
Remediation	To remediate failure of this policy test, create separate a partition for /home. **Creating a separate partition for /home**: 1. For new installations, check the box to "**Review and modify partitioning**" and create a separate partition for **/home**. 2. For systems that were previously installed, use the **Logical Volume Manager (LVM)** to create partitions. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

2.2.4.23 Verify That gpgcheck Is Globally Activated

Verify That gpgcheck Is Globally Activated

Description	The gpgcheck option, found in the main section of the /etc/yum.conf file determines if an RPM package's signature is always checked prior to its installation. It is important to ensure that an RPM's package signature is always checked prior to installation to ensure that the software is obtained from a trusted source.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/yum.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*gpgcheck=0[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) gpgcheck Option Deviation Does not exist
Remediation	To remediate failure of this policy test, set gpgcheck is globally activated. **Setting gpgcheck is globally activated:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/yum.conf** file. 3. Find the line **gpgcheck=<value>**. 4. If found, then set **<value>** to **1** and save the file. 5. If not found, then add the **gpgcheck=1** line under **[main]** section in **yum.conf** file and save it. For further details, please run command **man yum.conf** to read the manual page.

2.2.4.24 Verify That the AIDE Package Is Installed

Verify That the AIDE Package Is Installed

Description	Install AIDE to make use of the file integrity features to monitor critical files for changes that could affect the security of the system.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*aide-\d.*$/ (Flags:Multiline,Case insensitive,Comments mode) aide Package Exists
Remediation	To remediate failure of this policy test, install aide. **Installing aide**: 1. Become superuser or assume an equivalent role. 2. Install **aide** using **yum** command: **yum install <aide_pakage>** **Note:** The prelinking feature can interfere with AIDE because it alters binaries to speed up their start up times. Set **PRELINKING=no** in **/etc/sysconfig/prelink** and run **/usr/sbin/prelink -ua** to restore the binaries to their prelinked state, thus avoiding false positives from AIDE. For further details, please run the command **man yum** to read man page.

2.2.4.25 Verify That File Checking (AIDE) Is Implemented Periodically

Verify That File Checking (AIDE) Is Implemented Periodically

Description

Periodic file checking allows the system administrator to determine on a regular basis if critical files have been changed in an unauthorized fashion.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/var/spool/cron/root"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[^\#\n]*[\ \t]+/usr/sbin/aide[\ \t]+--check[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode)
Right Configuration Exists

Remediation

To remediate failure of this policy test, you should implement periodic file checking, in compliance with site policy.

Implementing periodic file checking:

1. Become superuser or assume an equivalent role.
2. Execute the following command:
   crontab -u root -e
3. Add the following line to the crontab:
   0 5 * * * /usr/sbin/aide --check
4. Save file to apply the change.

Note: The checking in this instance occurs every day at 5 am. Alter the frequency and time of the checks in compliance with site policy.

For further details, please run the command **man crontab** to read man page.

2.2.4.26 Verify That net.ipv4.conf.default.log_martians Is Equal to 1

Verify That net.ipv4.conf.default.log_martians Is Equal to 1

Description

This test checks that the net.ipv4.conf.default.log_martians parameter in /etc/sysctl.conf is set to true.
Enabling this feature and logging these packets allows an administrator to investigate the possibility that an attacker is sending spoofed packets to their server.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.default\.log_martians[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
net.ipv4.conf.default.log_martians Equals 1

Remediation

To remediate failure of this policy test, set the network parameter to log suspicious packets.

Setting the network parameter to log suspicious packets:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.default.log_martians = <value>**.
4. Set the line to **net.ipv4.conf.default.log_martians = 1** and save the file.
5. If the line is not found, add the line **net.ipv4.conf.default.log_martians = 1** and save the file.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

2.2.4.27 Verify That net.ipv4.conf.all.log_martians Is Equal to 1 (Default Value)

Verify That net.ipv4.conf.all.log_martians Is Equal to 1 (Default Value)

Description	This test checks that the net.ipv4.conf.all.log_martians parameter in /etc/sysctl.conf is set to true. Enabling this feature and logging these packets allows an administrator to investigate the possibility that an attacker is sending spoofed packets to their server.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/sysctl.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*net\.ipv4\.conf\.all\.log_martians[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode) net.ipv4.conf.all.log_martians Equals 1
Remediation	To remediate failure of this policy test, set the network parameter to log suspicious packets. **Setting the network parameter to log suspicious packets**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/sysctl.conf** file. 3. Find the line **net.ipv4.conf.all.log_martians = <value>**. 4. Set the line to **net.ipv4.conf.all.log_martians = 1** and save the file. 5. If the line is not found, add the line **net.ipv4.conf.all.log_martians = 1** and save the file. 6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings. 7. Run the **sysctl -p** command to apply the change. For further details, please run the command **man sysctl** to read man page, or refer to: https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

2.2.4.28 Verify That the ExecShield Feature Is Enabled

Verify That the ExecShield Feature Is Enabled

Description

Execshield is made up of a number of kernel features to provide protection against buffer overflow attacks. These features include prevention of execution in memory data space, and special handling of text buffers. Enabling any feature that can protect against buffer overflow attacks enhances the security of the system.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*kernel\.exec-shield[\ \t]*=[\ \t]*(\d+)[\ \t]*(?:\#|$)/ (Flags:Multiline,Comments mode)
kernel.exec-shield Equals 1

Remediation

To remediate failure of this policy test, set kernel.exec-shield to enable execshield in order to protect against buffer overflow attacks.

Setting kernel.exec-shield to enable:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **kernel.exec-shield = <value>**.
4. Set the **<value>** to **1** and save the file.
5. If there is no line setting **kernel.exec-shield**, add the following line:

    kernel.exec-shield = 1

   at the end of the file and it.
6. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl.conf** to read man page.

2.2.4.29 Verify That the Randomization Feature Is Enabled

Verify That the Randomization Feature Is Enabled

Description

Randomly placing virtual memory regions will make it difficult for to write memory page exploits as the memory placement will be consistently shifting.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*kernel\.randomize_va_space[\ \t]*=[\ \t]*(\d+)[\ \t]*(?:\#|$)/
(Flags:Multiline,Comments mode)
kernel.randomize_va_space Equals 2

Remediation

To remediate failure of this policy test, set kernel.randomize_va_space to enable randomized virtual memory region placement.

Set kernel.randomize_va_space to enable randomized virtual memory region placement:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the lines **kernel.randomize_va_space = <value>**.
4. Set the **<value>** to 2 and save the file.
5. If there no line setting **kernel.randomize_va_space**, add the following line:

   kernel.randomize_va_space = 2

   at the end of the file and save the file.
6. Reboot system to apply the change.

For further details, please run the command **man sysctl.conf** to read man page.

2.2.4.30 Verify That SELinux Is Not Disabled Using Grub Boot Loader

Verify That SELinux Is Not Disabled Using Grub Boot Loader

Description

SELinux must be enabled at boot time in grub.conf file to ensure that the controls it provides are not overwritten.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/boot/grub/grub.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*kernel\b[^\#]*[\ \t]selinux=0\b.*$/ (Flags:Multiline,Comments mode)
Disable selinux Mode in grub.conf File Does not exist

Remediation

To remediate failure of this policy test, enable SELinux in grub.conf file.

Enabling SELinux in grub.conf file:

1. Become superuser or assume an equivalent role.
2. Open the **/boot/grub/grub.conf** file
3. Remove **selinux=0** in the kernel line
4. Save file to apply the change.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/SELinux_Guide/rhlcommon-section-0016.html

2.2.4.31 Verify That the "Enforcing" Mode Is Not Disabled Using Grub Boot Loader

Verify That the "Enforcing" Mode Is Not Disabled Using Grub Boot Loader

Description	Enforcing is the default mode which will enable and enforce the SELinux security policy on the Linux. It will also deny unauthorized access and log actions in a log file.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/boot/grub/grub.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*kernel\b[^\#]*[\ \t]enforcing=0\b.*$/ (Flags:Multiline,Comments mode) enforcing Mode in grub.conf File Does not exist
Remediation	To remediate failure of this policy test, enable SELinux in grub.conf. **Enabling SELinux in grub.conf**: 1. Become superuser or assume an equivalent role. 2. Open the **/boot/grub/grub.conf** file 3. Remove **enforcing=0** in the kernel line 4. Save file to apply the change. For further details, please refer to: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/SELinux_Guide/rhlcommon-section-0016.html

2.2.4.32 Verify That SELinux Is Enabled at Boot Time

Verify That SELinux Is Enabled at Boot Time

Description	SELinux must be enabled at boot time in to ensure that the controls it provides are in effect at all times.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/selinux/config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^SELINUX=(?i:enforcing).*$/ (Flags:Multiline,Comments mode) Setting SELINUX to enforcing Exists
Remediation	To remediate failure of this policy test, ensure that SELinux is enabled at boot time. **To ensure that SELinux is enabled at boot time** : Become superuser or assume an equivalent role. 1. Open the **/etc/selinux/config** file. 2. Find the line **SELINUX=<parameter>**. 3. If found, then set **<parameter>** to **enforcing** and save the file. 4. If not found, then add the **SELINUX=enforcing** line to the file and save it. 5. Reboot to apply the change. For further details, please refer to: RHEL 5, 6: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/Reference_Guide/s2-SELinux-files-etc.html RHEL 7: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/SELinux_Users_and_Administrators_Guide/

2.2.4.33 Verify That SELinux Is Running

Verify That SELinux Is Running

Description SELinux must be enabled to ensure that the controls it provides are in effect at all times.

Severity 0

Weight 5

Type Content Test

Rules Get SELinux State

Excluded Nodes Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element Equals "Get SELinux State"

Version conditions If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*SELinux[\ \t]+status:[\ \t]+enabled[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode)
SELinux Enabled Exists

Remediation To remediate failure of this policy test, ensure that SELinux is running.

To ensure that SELinux is running on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/grub.conf** file
3. Remove **selinux=0** and **enforcing=0** parameter in the kernel line
4. Save file to apply the change.
5. Open the **/etc/selinux/config** file.
6. Find the line **SELINUX=<parameter>**.
7. If found, then set **<parameter>** to **enforcing** and save the file.
8. If not found, then add the **SELINUX=enforcing** line to the file and save it.
9. Reboot to apply change.

To ensure that SELinux is running on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/default/grub** file.
3. Remove **selinux=0** and **enforcing=0** parameter in the **GRUB_CMDLINE_LINUX="parameter1 parameter2 ..."** line.
4. Run **grub2-mkconfig -o /boot/grub2/grub.cfg** command to apply the change.
5. Open the **/etc/selinux/config** file.
6. Find the line **SELINUX=<parameter>**.
7. If found, then set **<parameter>** to **enforcing** and save the file.
8. If not found, then add the **SELINUX=enforcing** line to the file and save it.
9. Reboot to apply change.

For further details, please refer to:

RHEL 5, 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/Reference_Guide/s2-SELinux-files-etc.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/SELinux_Users_and_Administrators_Guide/

2.2.4.34 Verify That crond Daemon Is Enabled

Verify That crond Daemon Is Enabled

Description	The crond daemon is used to execute batch jobs on the system. While there may not be user jobs that need to be run on the system, the system does have maintenance jobs that may include security monitoring that have to run and cron is used to execute them.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*crond[\ \t]+.*\bon\b.*/ (Flags:Multiline,Comments mode) crond Daemon Enabled Exists
Remediation	To remediate failure of this policy test, turn on the crond daemon. **Turning on the crond daemon:** 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list crond** command to check the service status. 3. Turn on the **crond** daemon using the **chkconfig crond on** command. For further details, please run the command **man chkconfig** to read man page.

2.2.4.35 Verify That the /etc/ssh/sshd_config File Contains 'MaxAuthTries'

Verify That the /etc/ssh/sshd_config File Contains 'MaxAuthTries'

Description	This test verifies that the /etc/ssh/sshd_config file contains 'MaxAuthTries'. The MaxAuthTries option determines the maximum number of login attempts per connection. MaxAuthTries should be greater than 0 and less than or equal to 4.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*MaxAuthTries[\ \t]+(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) SSH MaxAuthTries Greater than 0 AND SSH MaxAuthTries Less than or equal 4
Remediation	To remediate failure of this policy test, limit the maximum number of authentication attempts which are permitted per connection at 4. **Limiting the maximum number of authentication attempts which are permitted per connection at 4:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line that contains **MaxAuthTries <value>**. 4. Set **<value>** to **4** or less and greater than **0** then save the file. 5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.

2.2.4.36 Verify That SELinux Meets or Exceeds the Default Targeted Policy

Verify That SELinux Meets or Exceeds the Default Targeted Policy

Description

Security configuration requirements vary from site to site. Some sites may mandate a policy that is stricter than the default policy, which is perfectly acceptable. This item is intended to ensure that at least the default recommendations are met.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/selinux/config"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^SELINUXTYPE=[\ \t]*targeted[\ \t]*$/ (Flags:Multiline,Comments mode)
SELinux Type Exists

Remediation

To remediate failure of this policy test, ensure that SELinux is enabled at boot time.

To ensure that SELinux is enabled at boot time:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/selinux/config** file.
3. Find the line **SELINUXTYPE=<parameter>**.
4. If found, then set **<parameter>** to **targeted** and save the file.
5. If not found, then add the **SELINUXTYPE=targeted** line to the file and save it.
6. Reboot to apply the change.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security-Enhanced_Linux/sect-Security-Enhanced_Linux-Working_with_SELinux-Enabling_and_Disabling_SELinux.html

2.2.4.37 Password Protect Enabled Accounts

Password Protect Enabled Accounts

Description

This test determines if accounts are allowed to have empty passwords. All accounts should have strong passwords or the account should be locked.

Severity

0

Weight

5

Type

Content Test

Rules

Get All Configurations in system-auth File

Element

Equals "Get All Configurations in system-auth File"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[^\#]*\bnullok\b.*/ (Flags:Multiline,Comments mode)
Password Protect Setting Does not exist

Remediation

To remediate failure of this policy test, remove entries for nullok in the /etc/pam.d/system-auth file.

Removing entries for nullok in the /etc/pam.d/system-auth file:

1. Become superuser or assume an equivalent role.
2. Run following command to list all files that should be updated:

```
PamConfigFile="/etc/pam.d/system-auth"; directory="/etc/
pam.d"; GetIncludeConfig(){ Files="";FileConfig=$1;Mod
uleInterface=$2;IncludeFiles=`/bin/cat "$FileConfig" 2>/dev/
null | /bin/awk -F"#" ' $1 ~ /^[[:space:]]*('$ModuleInterfac
e')[[:space:]]+(required|requisite)[[:space:]]+(.*\/)?pam_s
tack\.so[[:space:]]+service=\w+/ {print $1}' | /bin/awk -F"ser
vice=" '{print $2}' | /bin/awk 'BEGIN {ORS=";"} {print $1}'`
;Files=$Files$IncludeFiles; IncludeFiles=`/bin/cat "$FileCon
fig" 2>/dev/null | /bin/awk -F"#" ' $0 ~ /^[[:space:]]*('$Modu
leInterface')[[:space:]]+include[[:space:]].*/ {print $1}' | /bin/
awk 'BEGIN {ORS=";"} {print $3}'`;  Files=$Files$IncludeFiles;
SaveIFS=$IFS; IFS=";";for file in $Files; do if [ "`/usr/bin/
dirname $file 2>/dev/null`" != "." ]; then if [ -e "$file" ]; then /
bin/echo $file 2>/dev/null ;  GetIncludeConfig  "$file" "$Mod
uleInterface";fi; else full_path=$directory"/"$file; if [ -e "$ful
l_path" ]; then /bin/echo $full_path ;GetIncludeConfig  "$ful
l_path" "$ModuleInterface";    fi; fi; done;IFS=$SaveIFS; }
;/bin/echo "$PamConfigFile" 2>/dev/null; GetIncludeConfig
"$PamConfigFile" "auth";GetIncludeConfig "$PamConfigFile"
"password"; GetIncludeConfig "$PamConfigFile" "session";
GetIncludeConfig "$PamConfigFile" "account";
```

3. For each file that listed in step 2, run the **grep nullok <file>** command to list entries for **nullok**.
4. Open the file.
5. Remove **nullok** contained in entries found in step 3 and save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-pam-sample-simple.html

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/pam.d/system-auth"
RegEx="[[:space:]]+nullok\>"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to remove null ok
UpdateLog=`(/bin/awk -F "#" 'BEGIN{OFS="#"}\
        $1 ~ /'$RegEx'/ {
            gsub(/'$RegEx'/,"",$1)
        } {print}' "$BackupName" > "$FileName") 2>&1`

if [ -n "$UpdateLog" ]; then
    /bin/echo "FAILURE-7001: Could not remove [nullok]"\
        "option in [$FileName] file"
    # Rollback to the original file
    /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
    exit 7001
fi

/bin/echo "SUCCESS-7001: [nullok] option removed in [$FileName]
 file"
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003882
# AR_TEST_NAME = Password Protect Enabled Accounts
```

Post Remediation Category	*None*
Remediated Elements	/etc/pam.d/system-auth /etc/pam.d/system-auth-ac
Post Remediation Steps	No additional Post Remediation steps

2.2.4.38 Verify That the /var/tmp Directory Is Bound to the /tmp Directory in /etc/fstab

Verify That the /var/tmp Directory Is Bound to the /tmp Directory in /etc/fstab

Description	The /var/tmp directory is normally a standalone directory in the /var file system. Binding /var/tmp to /tmp establishes an unbreakable link to /tmp that cannot be removed (even by the root user). It also allows /var/tmp to inherit the same mount options that /tmp owns, allowing /var/tmp to be protected in the same /tmp is protected. It will also prevent /var from filling up with temporary files as the contents of /var/tmp will actually reside in the file system containing /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*/tmp[\ \t]+/var/tmp[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bbind\b.*$/ (Flags:Multiline,Comments mode) Right Configuration Exists
Remediation	To remediate failure of this policy test, bind mount the /var/tmp directory to /tmp. **Binding mount the /var/tmp directory to /tmp**: 1. Become superuser or assume an equivalent role. 2. Run following command: **mount --bind /tmp /var/tmp** 3. Open the **/etc/fstab** file. 4. Edit the file to contain the following line: **/tmp /var/tmp none bind 0 0** 5. Save file to apply the change. For further details, please run the command **man fstab** to read man page.

2.2.4.39 Verify That the /var/tmp Directory Is Bound to the /tmp Directory

Verify That the /var/tmp Directory Is Bound to the /tmp Directory

Description	The /var/tmp directory is normally a standalone directory in the /var file system. Binding /var/tmp to /tmp establishes an unbreakable link to /tmp that cannot be removed (even by the root user). It also allows /var/tmp to inherit the same mount options that /tmp owns, allowing /var/tmp to be protected in the same /tmp is protected. It will also prevent /var from filling up with temporary files as the contents of /var/tmp will actually reside in the file system containing /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*/tmp[\ \t]+on[\ \t]+/var/tmp[\ \t]+type[\ \t]+\S+[\ \t]+\(.*\bbind\b.*\).*$/ (Flags:Multiline,Comments mode) Right Configuration Exists
Remediation	To remediate failure of this policy test, bind mount the /var/tmp directory to /tmp. **Binding mount the /var/tmp directory to /tmp**: 1. Become superuser or assume an equivalent role. 2. Run following command: **mount --bind /tmp /var/tmp** 3. Open the **/etc/fstab** file. 4. Edit the file to contain the following line: **/tmp /var/tmp none bind 0 0** 5. Save file to apply the change. For further details, please run the command **man fstab** to read man page.

2.2.4.40 Verify That the noexec Option Is Added to Removable Media Partitions

Verify That the noexec Option Is Added to Removable Media Partitions

Description

This test determines if the 'noexec' option is configured for removable media. Allowing users to execute binaries from removable media such as USB keys exposes the system to potential compromise.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/fstab"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*[^\#\s\n]+[\ \t]+[^\#\s\n]+[\ \t]+(?:iso9660|udf|usbfs)[\ \t]+(?:(?!\bnoexec\b)[^\#\n&&\S])+[\ \t]+.*$/ (Flags:Multiline,Comments mode)
Removable Media without noexec Option Does not exist

Remediation

To remediate failure of this policy test, add the noexec option to removable media.

Adding the noexec option to removable media:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/fstab** file.
3. Find the line(s) with **iso9660**, **udf** and **usbfs** filesystem type for removable media.
4. If the line does not contain **noexec** option, add the noexec option to the fourth field, using a comma to separate from other options.
5. Reload **fstab** by using the **mount -a** command.

2.2.4.41 Verify That net.ipv4.conf.default.secure_redirects Is Equal to 0

Verify That net.ipv4.conf.default.secure_redirects Is Equal to 0

Description

This test checks that the net.ipv4.conf.default.secure_redirects parameter in /etc/sysctl.conf is set to false.
Disabling ICMP redirect messages for gateways listed in the default gateway list reduces the risk of source address spoofing.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.default\.secure_redirects[\ \t]*=[\ \t]*(\d+)[\ \t]*$/
(Flags:Multiline,Comments mode)
net.ipv4.conf.default.secure_redirects Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to reject ICMP redirect messages for gateways listed in the default gateway list.

Setting the network parameter to reject ICMP redirect messages for gateways listed in the default gateway list:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.default.secure_redirects = <value>**.
4. Set the line to **net.ipv4.conf.default.secure_redirects = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.default.secure_redirects**, add the following line:

   net.ipv4.conf.default.secure_redirects = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.default.secure_redirects"
Regex="net\.ipv4\.conf\.default\.secure_redirects"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000888
# AR_TEST_NAME = Verify That
 net.ipv4.conf.default.secure_redirects Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category Other

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.42 Verify That net.ipv4.conf.all.accept_source_route Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.all.accept_source_route Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.all.accept_source_route parameter in /etc/sysctl.conf is set to false.
Disabling this setting helps to prevent exploitation of IP trust relationships with spoofed source packets.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*net\.ipv4\.conf\.all\.accept_source_route[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
(The net.ipv4.conf.all.accept_source_route Setting Exists AND net.ipv4.conf.all.accept_source_route Equals 0) OR
The net.ipv4.conf.all.accept_source_route Setting Does not exist

Remediation

To remediate failure of this policy test, set the network parameter to only accept routing header type 2 on all interfaces.

Setting the network parameter to only accept routing header type 2 on all interfaces:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.all.accept_source_route = <value>**.
4. Set the line **net.ipv4.conf.all.accept_source_route = 0** and save the file.
5. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
6. Run the **sysctl -p** command to apply the change.

For further details, please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.all.accept_source_route"
Regex="net\.ipv4\.conf\.all\.accept_source_route"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000889
# AR_TEST_NAME = Verify That
 net.ipv4.conf.all.accept_source_route Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

`Other`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.43 Verify That net.ipv4.conf.default.accept_source_route Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.default.accept_source_route Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.default.accept_source_route parameter in /etc/sysctl.conf is set to false.
Source routing can be used in IP spoofing attacks and should be disabled.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.default\.accept_source_route[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
net.ipv4.conf.default.accept_source_route Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to only accept routing header type 2 on the default interface.

Setting the network parameter to only accept routing header type 2 on the default interface:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.default.accept_source_route = <value>**.
4. Set the line **net.ipv4.conf.default.accept_source_route = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.default.accept_source_route**, add the following line:

   net.ipv4.conf.default.accept_source_route = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.default.accept_source_route"
Regex="net\.ipv4\.conf\.default\.accept_source_route"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000892
# AR_TEST_NAME = Verify That
 net.ipv4.conf.default.accept_source_route Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category Other

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.44 Verify That net.ipv4.conf.all.secure_redirects Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.all.secure_redirects Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.all.secure_redirects parameter in /etc/sysctl.conf is set to false.
Disabling ICMP redirect messages for gateways listed in the default gateway list reduces the risk of source address spoofing.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.all\.secure_redirects[\ \t]*=[\ \t]*(\d+)[\ \t]*$/
(Flags:Multiline,Comments mode)
net.ipv4.conf.all.secure_redirects Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to reject ICMP redirect messages for gateways listed in the default gateway list on all interfaces.

Setting the network parameter to reject ICMP redirect messages for gateways listed in the default gateway list on all interfaces:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.all.secure_redirects = <value>**.
4. Set the line **net.ipv4.conf.all.secure_redirects = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.all.secure_redirects**, add the following line:

   net.ipv4.conf.all.secure_redirects = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.all.secure_redirects"
Regex="net\.ipv4\.conf\.all\.secure_redirects"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000893
# AR_TEST_NAME = Verify That net.ipv4.conf.all.secure_redirects
 Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category	Other
Remediated Elements	*None*
Post Remediation Steps	To complete this remediation: 1. Become superuser or assume an equivalent role. 2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.45 Verify That net.ipv4.conf.all.accept_redirects Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.all.accept_redirects Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.all.accept_redirects parameter in /etc/sysctl.conf is set to false.
For hosts with correct routing tables, ICMP redirect messages are considered malicious. For hosts with incorrect routing tables, ignoring these packets will only slightly impact net work performance.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.all\.accept_redirects[\ \t]*=[\ \t]*(\d+)[\ \t]*$/
(Flags:Multiline,Comments mode)
net.ipv4.conf.all.accept_redirects Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to reject ICMP redirect messages on all interfaces.

Setting the network parameter to reject ICMP redirect messages on all interfaces:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.all.accept_redirects = <value>**.
4. Set the line to **net.ipv4.conf.all.accept_redirects = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.all.accept_redirects**, add the following line:

   net.ipv4.conf.all.accept_redirects = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.all.accept_redirects"
Regex="net\.ipv4\.conf\.all\.accept_redirects"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000894
# AR_TEST_NAME = Verify That net.ipv4.conf.all.accept_redirects
 Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

Other

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.46 Verify That net.ipv4.conf.all.rp_filter Is Equal to 1

Verify That net.ipv4.conf.all.rp_filter Is Equal to 1

Description

This test checks that the net.ipv4.conf.all.rp_filter parameter in /etc/sysctl.conf is set to true.
Enabling this setting helps to identify spoofed source addresses.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.all\.rp_filter[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
net.ipv4.conf.all.rp_filter Equals 1

Remediation

To remediate failure of this policy test, configure the network parameter to enable a reverse-path filter for IPv4 network traffic when possible on all interfaces.

Configuring the network parameter to enable a reverse-path filter for IPv4 network traffic when possible on all interfaces:

1. Become superuser or assume an equivalent role.
2. Open **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.all.rp_filter = <value>**.
4. Set the line to **net.ipv4.conf.all.rp_filter = 1** and save the file.
5. If there is no line setting **net.ipv4.conf.all.rp_filter**, add the following line:

   net.ipv4.conf.all.rp_filter = 1

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.all.rp_filter"
Regex="net\.ipv4\.conf\.all\.rp_filter"
SeparateSymbol="="
Value="1"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000897
# AR_TEST_NAME = Verify That net.ipv4.conf.all.rp_filter Is Equal
 to 1


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category Other

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.47 Verify That net.ipv4.conf.default.rp_filter Is Equal to 1

Verify That net.ipv4.conf.default.rp_filter Is Equal to 1

Description	This test checks that the net.ipv4.conf.default.rp_filter parameter in /etc/sysctl.conf is set to true. Enabling reverse path source validation for the default interface will help to identify spoofed source addresses.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/sysctl.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*net\.ipv4\.conf\.default\.rp_filter[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode) net.ipv4.conf.default.rp_filter Equals 1
Remediation	To remediate failure of this policy test, set the network parameter to enable reverse path source validation on the default interface. **Setting the network parameter to enable reverse path source validation on the default interface**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/sysctl.conf** file. 3. Find the line **net.ipv4.conf.default.rp_filter = <value>**. 4. Set the line to **net.ipv4.conf.default.rp_filter = 1** and save the file. 5. If there is no line setting **net.ipv4.conf.default.rp_filter**, add the following line: **net.ipv4.conf.default.rp_filter = 1** at the end of the file and save it. 6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings. 7. Run the **sysctl -p** command to apply the change. For further details, please run the command **man sysctl** to read man page, or please refer to: https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.default.rp_filter"
Regex="net\.ipv4\.conf\.default\.rp_filter"
SeparateSymbol="="
Value="1"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000898
# AR_TEST_NAME = Verify That net.ipv4.conf.default.rp_filter Is
 Equal to 1


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

Other

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.48 Verify That net.ipv4.tcp_syncookies Is Equal to 1 (Default Value)

Verify That net.ipv4.tcp_syncookies Is Equal to 1 (Default Value)

Description

This test verifies that the system is configured to use TCP syncookies. Syncookies can be used to track a connection when a subsequent ACK is received, verifying the initiator is attempting a valid connection and is not a flood source. This feature is activated when a flood condition is detected, and enables the system to continue servicing valid connection requests.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.tcp_syncookies[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
net.ipv4.tcp_syncookies Equals 1

Remediation

To remediate failure of this policy test, set the network parameter to enable sending of syncookies to help prevent syn flood attacks.

Setting the network parameter to enable sending of syncookies to help prevent syn flood attacks:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.tcp_syncookies = <value>**.
4. Set the line to **net.ipv4.tcp_syncookies = 1** and save the file.
5. If there is no line setting **net.ipv4.tcp_syncookies**, add the following line:

   net.ipv4.tcp_syncookies = 1

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.tcp_syncookies"
Regex="net\.ipv4\.tcp_syncookies"
SeparateSymbol="="
Value="1"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000887
# AR_TEST_NAME = Verify That net.ipv4.tcp_syncookies Is Equal to
 1


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category	Other
Remediated Elements	*None*
Post Remediation Steps	To complete this remediation: 1. Become superuser or assume an equivalent role. 2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.49 Verify That net.ipv4.icmp_echo_ignore_broadcasts Is Equal to 1 (Default Value)

Verify That net.ipv4.icmp_echo_ignore_broadcasts Is Equal to 1 (Default Value)

Description

This test checks that the net.ipv4.icmp_echo_ignore_broadcasts parameter in /etc/sysctl.conf is set to true.
This allows the system to continue to respond to normal ping packets while preventing it from participating in creating floods of echo replies.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*net\.ipv4\.icmp_echo_ignore_broadcasts[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
(The net.ipv4.icmp_echo_ignore_broadcasts Setting Exists AND
net.ipv4.icmp_echo_ignore_broadcasts Equals 1) OR
The net.ipv4.icmp_echo_ignore_broadcasts Setting Does not exist

Remediation

To remediate failure of this policy test, set the network parameter to ignore all ICMP ECHO and TIMESTAMP requests sent via broadcast/multicast.

Setting the network parameter to ignore all ICMP ECHO and TIMESTAMP requests sent via broadcast/multicast:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.icmp_echo_ignore_broadcasts = <value>**.
4. Set the line to **net.ipv4.icmp_echo_ignore_broadcasts = 1** and save the file.
5. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
6. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.icmp_echo_ignore_broadcasts"
Regex="net\.ipv4\.icmp_echo_ignore_broadcasts"
SeparateSymbol="="
Value="1"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000895
# AR_TEST_NAME = Verify That net.ipv4.icmp_echo_ignore_broadcasts
 Is Equal to 1


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

`Other`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.50 Verify That net.ipv4.conf.default.accept_redirects Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.default.accept_redirects Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.default.accept_redirects parameter in /etc/sysctl.conf is set to false.
For hosts with correct routing tables, ICMP redirect messages are considered malicious.
For hosts with incorrect routing tables, ignoring these packets will only slightly impact network performance.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.default\.accept_redirects[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
net.ipv4.conf.default.accept_redirects Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to reject ICMP redirects on the default interface.

Setting the network parameter to reject ICMP redirects on the default interface:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.default.accept_redirects = <value>**.
4. Set the line to **net.ipv4.conf.default.accept_redirects = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.default.accept_redirects**, add the following line:

   net.ipv4.conf.default.accept_redirects = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl** to read man page, or refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.default.accept_redirects"
Regex="net\.ipv4\.conf\.default\.accept_redirects"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000896
# AR_TEST_NAME = Verify That
 net.ipv4.conf.default.accept_redirects Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

`Other`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.51 Verify That net.ipv4.conf.default.send_redirects Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.default.send_redirects Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.default.send_redirects parameter in /etc/sysctl.conf is set to false.
Send redirects can have a negative impact on network performance and efficiency and should be disabled.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.default\.send_redirects[\ \t]*=[\ \t]*(\d+)[\ \t]*$/
(Flags:Multiline,Comments mode)
net.ipv4.conf.default.send_redirects Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to disable sending redirect messages for the default interface.

Setting the network parameter to disable sending redirect messages for the default interface:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.default.send_redirects = <value>**.
4. Set the line to **net.ipv4.conf.default.send_redirects = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.default.send_redirects**, add the following line:

   net.ipv4.conf.default.send_redirects = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.default.send_redirects"
Regex="net\.ipv4\.conf\.default\.send_redirects"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"'"$SeparateSymbol"'"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000899
# AR_TEST_NAME = Verify That net.ipv4.conf.default.send_redirects
 Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

`Other`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.52 Verify That net.ipv4.conf.all.send_redirects Is Equal to 0 (Default Value)

Verify That net.ipv4.conf.all.send_redirects Is Equal to 0 (Default Value)

Description

This test checks that the net.ipv4.conf.all.send_redirects parameter in /etc/sysctl.conf is set to false.
Send redirects can have a negative impact on network performance and efficiency and should be disabled.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*net\.ipv4\.conf\.all\.send_redirects[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
net.ipv4.conf.all.send_redirects Equals 0

Remediation

To remediate failure of this policy test, set the network parameter to disable sending redirects messages for all interfaces.

Setting the network parameter to disable sending redirects messages for all interfaces:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.conf.all.send_redirects = <value>**.
4. Set the line to **net.ipv4.conf.all.send_redirects = 0** and save the file.
5. If there is no line setting **net.ipv4.conf.all.send_redirects**, add the following line:

   net.ipv4.conf.all.send_redirects = 0

   at the end of the file and save it.
6. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
7. Run the **sysctl -p** command to apply the change.

For further details, please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.conf.all.send_redirects"
Regex="net\.ipv4\.conf\.all\.send_redirects"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000900
# AR_TEST_NAME = Verify That net.ipv4.conf.all.send_redirects Is
 Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category Other

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.53 Verify That net.ipv4.ip_forward Is Equal to 0

Verify That net.ipv4.ip_forward Is Equal to 0

Description

This test checks that the net.ipv4.ip_forward parameter in /etc/sysctl.conf is set to false. Forwarding packets between interfaces is considered a slight network security risk and should be disabled.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*net\.ipv4\.ip_forward[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
(The net.ipv4.ip_forward Seting Exists AND
net.ipv4.ip_forward Equals 0) OR
The net.ipv4.ip_forward Seting Does not exist

Remediation

To remediate failure of this policy test, set the network parameter to disable IP forwarding.

Setting the network parameter to disable IP forwarding:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.ip_forward = <value>**.
4. Set the **<value>** to **0** and save the file.
5. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
6. Run the **sysctl -p** command to apply the change.

For further details, please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.ip_forward"
Regex="net\.ipv4\.ip_forward"
SeparateSymbol="="
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0000901
# AR_TEST_NAME = Verify That net.ipv4.ip_forward Is Equal to 0


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category	Other
Remediated Elements	*None*

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.54 Verify That net.ipv4.icmp_ignore_bogus_error_responses Is Equal to 1 (Default Value)

Verify That net.ipv4.icmp_ignore_bogus_error_responses Is Equal to 1 (Default Value)

Description

Some routers violate RFC1122 by sending bogus responses to broadcast frames. Such violations are normally logged via a kernel warning. If this is set to true (set to 1), the kernel will not give such warnings, which will avoid log file clutter. This test checks that the net.ipv4.icmp_ignore_bogus_error_responses parameter in /etc/sysctl.conf is set to true (set to 1).

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysctl.conf"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*net\.ipv4\.icmp_ignore_bogus_error_responses[\ \t]*=[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
(The net.ipv4.icmp_ignore_bogus_error_responses Setting Exists AND
net.ipv4.icmp_ignore_bogus_error_responses Equals "1") OR
The net.ipv4.icmp_ignore_bogus_error_responses Setting Does not exist

Remediation

To remediate failure of this policy test, set the network parameter to ignore ICMPv4 bogus error responses.

Setting the network parameter to ignore ICMPv4 bogus error responses:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the line **net.ipv4.icmp_ignore_bogus_error_responses = <value>**.
4. Set the line to **net.ipv4.icmp_ignore_bogus_error_responses = 1** and save the file.
5. Run the **/sbin/sysctl -w net.ipv4.route.flush=1** command to flush all the tcp cache settings.
6. Run the **sysctl -p** command to apply the change.

For further details, please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-proc-sysctl.html

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/sysctl.conf"
ParameterName="net.ipv4.icmp_ignore_bogus_error_responses"
Regex="net\.ipv4\.icmp_ignore_bogus_error_responses"
SeparateSymbol="="
Value="1"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]*$/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($1 ~ /^[[:space:]]*'"$Regex"'[[:space:]]*$/) {
        $0 = "'"$ParameterName"'"'"$SeparateSymbol"'"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_WILDCARD_SETTING
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0014014
# AR_TEST_NAME = Verify That
 net.ipv4.icmp_ignore_bogus_error_responses Is Equal to 1


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>sysctl -p</b> command to
 reload settings in the <b>/etc/sysctl.conf </b> file.</li></ol>
```

Post Remediation Category

`Other`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **sysctl -p** command to reload settings in the **/etc/sysctl.conf** file.

2.2.4.55 Verify PASS_MIN_DAYS Parameter in /etc/login.defs

Verify PASS_MIN_DAYS Parameter in /etc/login.defs

Description	This test verifies that /etc/login.defs is configured to prevent password changes for at least 7 days. This setting is used for the creation of new accounts. Preventing frequent password resets helps protect against brute-force password cracking programs.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/login.defs"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*PASS_MIN_DAYS[\ \t]+(\d+)[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode) PASS_MIN_DAYS Greater than or equal 7
Remediation	To remediate failure of this policy test, set the minimum number of days allowed between password changes to at least 7. **Setting the minimum number of days allowed between password changes to at least 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/login.defs** file. 3. Find the line **PASS_MIN_DAYS <value>**. 4. Set the **<value>** to **7** or greater and save the file. For further details, please run the command **man login.defs** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/login.defs"
ParameterName="PASS_MIN_DAYS"
SeparateSymbol=" "
Value="7"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003382
# AR_TEST_NAME = Verify PASS_MIN_DAYS Parameter in /etc/
login.defs
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

2.2.4.56 Verify That No Legacy '+' Entries Exist in /etc/passwd

Verify That No Legacy '+' Entries Exist in /etc/passwd

Description	This test verifies that no legacy '+' entries exist in /etc/passwd. At one time, '+' entries were employed as markers for systems to insert data from NIS maps. These entries can serve as a way for attackers to gain privileged access on the system, and should be removed if they exist.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/passwd"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*\+.*/ (Flags:Multiline,Case insensitive,Comments mode) Legacy '+' Entries Does not exist
Remediation	To remediate failure of this policy test, remove or comment out legacy '+' entries in the /etc/passwd file. **Removing or commenting out legacy '+' entries in the /etc/passwd file**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/passwd** file. 3. Find lines those contain the plus signs at the beginning. 4. Remove or comment out lines and save the file.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/passwd"

# Check if + entry exists
IsExisted=`/bin/egrep "^[[:space:]]*\+" $FileName 2>/dev/null`

if [ -z "$IsExisted" ]; then
    /bin/echo "SUCCESS-7001: There is no [+] entry in [$FileName]
 file"
    exit 0
fi

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to remove [+] entry
RemovedEntry=`(/bin/awk '$1 !~ /^\+/  \
    {print}' "$BackupName" > "$FileName") 2>&1`

if [ -n "$RemovedEntry" ]; then
    /bin/echo "FAILURE-7001: Could not remove [+]"\
        "entry in [$FileName] file"
    # Rollback to the original file
    /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
    exit 7001
fi

/bin/echo "SUCCESS-7001: Removed [+] entry in [$FileName] file"
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003386
# AR_TEST_NAME = Verify That No Legacy '+' Entries Exist in /etc/
passwd
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

2.2.4.57 Verify That No Legacy '+' Entries Exist in /etc/group

Verify That No Legacy '+' Entries Exist in /etc/group

Description

This test verifies that no legacy '+' entries exist in /etc/group. At one time, '+' entries were employed as markers for systems to insert data from NIS maps. These entries can serve as a way for attackers to gain privileged access on the system, and should be removed if they exist.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/group"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*\+.*/ (Flags:Multiline,Case insensitive,Comments mode)
Legacy '+' Entries Does not exist

Remediation

To remediate failure of this policy test, remove or comment out legacy '+' entries in the /etc/group file.

Removing or commenting out legacy '+' entries in the /etc/group file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/group** file.
3. Find lines those contains the plus signs at the beginning.
4. Remove or comment out lines and save the file.

For further details, please run the command **man gpasswd** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/group"

# Check if + entry exists
IsExisted=`/bin/egrep "^[[:space:]]*\+" $FileName 2>/dev/null`

if [ -z "$IsExisted" ]; then
    /bin/echo "SUCCESS-7001: There is no [+] entry in [$FileName]
 file"
    exit 0
fi

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to remove entry
RemovedEntry=`(/bin/awk '$1 !~ /^\+/ \
    {print}' "$BackupName" > "$FileName") 2>&1`

if [ -n "$RemovedEntry" ]; then
    /bin/echo "FAILURE-7001: Could not remove [+]"\
        "entry in [$FileName] file"
    # Rollback to the original file
    /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
    exit 7001
fi

/bin/echo "SUCCESS-7001: Removed [+] entry in [$FileName] file"
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003387
# AR_TEST_NAME = Verify That No Legacy '+' Entries Exist in /etc/
group
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

2.2.4.58 Verify That No Legacy '+' Entries Exist in /etc/shadow

Verify That No Legacy '+' Entries Exist in /etc/shadow

Description	This test verifies that no legacy '+' entries exist in /etc/shadow. At one time, '+' entries were employed as markers for systems to insert data from NIS maps. These entries can serve as a way for attackers to gain privileged access on the system, and should be removed if they exist.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/shadow"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*\+.*/ (Flags:Multiline,Case insensitive,Comments mode) Legacy '+' Entries Does not exist
Remediation	To remediate failure of this policy test, remove or comment out legacy '+' entries in the /etc/shadow file. **Removing or commenting out legacy '+' entries in the /etc/shadow file**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/shadow** file. 3. Find lines those contains the plus signs at the beginning. 4. Remove or comment out lines and save the file.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/shadow"

# Check if + entry exists
IsExisted=`/bin/egrep "^[[:space:]]*\+" $FileName 2>/dev/null`

if [ -z "$IsExisted" ]; then
    /bin/echo "SUCCESS-7001: There is no [+] entry in [$FileName]
 file"
    exit 0
fi

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to remove entry
CommentEntry=`(/bin/awk '$1 !~ /^\+/ \
    {print}' "$BackupName" > "$FileName") 2>&1`

if [ -n "$CommentEntry" ]; then
    /bin/echo "FAILURE-7001: Could not remove [+]"\
        "entry in [$FileName] file"
    # Rollback to the original file
    /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
    exit 7001
fi

/bin/echo "SUCCESS-7001: Removed [+] entry in [$FileName] file"
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003388
# AR_TEST_NAME = Verify That No Legacy '+' Entries Exist in /etc/
shadow
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

2.2.4.59 Verify PASS_MIN_DAYS Setting for Non-system Accounts

Verify PASS_MIN_DAYS Setting for Non-system Accounts

Description

This test verifies that all non-system accounts are configured to prevent password changes for at least 7 days.
Preventing frequent password resets helps protect against brute-force password cracking programs.

Severity

0

Weight

5

Type

Content Test

Rules

Verify Expiration Password for Non-system Account

Element

Equals "Expiration Password"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^\S+:.*PASS_MIN_DAYS=[\ \t]*(?:|-\d+|0*[1-6]?)[\ \t]+.*/ (Flags:Multiline,Comments mode)
'Fail Minimum Password Age' for Non-system Accounts Does not exist

Remediation

To remediate failure of this policy test, set the minimum number of days between password changes to at least 7 for non-system accounts.

Setting the minimum number of days between password changes to at least 7 for non-system accounts:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   for Acc in `awk -F: '$1 !~ /^[[:space:]]*#/ && $3>=500 && $3!=65534 {print $1}' /etc/passwd 2>/dev/null`; do awk -F: '$1 ~ /^[[:space:]]*'$Acc'$/ && $2!~/[!*]+/ && ($4<7 || $4 ~ /^[[:space:]]*$/) {print $1" account has PASS_MIN_DAYS="$4}' /etc/shadow 2>/dev/null; done
   ```

   to list non-system accounts of which the minimum number of days between password changes is less than **7**.
3. Change the minimum number of days between password changes to at least **7** for non-system accounts found in step 2 using the **chage -m <value> <user_name>** command, where **<value>** is greater than or equal to **7**.

For further details, please run the command **man chage** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
PasswordParameter="PASS_MIN_DAYS"
Value="7"
FailedAccounts=`/bin/awk -F":" '$1 !~ /[[:space:]]*#/ && $2!~/[!
*]+/ {
    GetIdCmd="/usr/bin/id -u " $1 " 2>/dev/null"; Uid=""
    GetIdCmd | getline Uid
    if(Uid ~ /^[0-9]+$/ && 0+Uid >= 500 && 0+Uid < 65534){
        if($4 !~ /^-?[0-9]+$/ || 0+$4 < 7){ print $1 }
    }
}' /etc/shadow 2>/dev/null`

# Issue the command to change PASS_MIN_DAYS setting for non-
system accounts
SavedIFS=$IFS
IFS=`/bin/echo -ne "\n\b"`

if [ -n "${FailedAccounts}" ]; then
    for Account in $FailedAccounts; do
        UpdateLog=`/usr/bin/chage -m $Value $Account 2>&1`
        if [ -n "$UpdateLog" ]; then
            FailureUpdate=`[ -z "$FailureUpdate" ] ||\
                /bin/echo $FailureUpdate"\n"`$Account
        else
            SuccessUpdate=`[ -z "$SuccessUpdate" ] ||\
                /bin/echo $SuccessUpdate"\n"`$Account
        fi
    done
else
    /bin/echo "SUCCESS-7001: No account with failure
 [$PasswordParameter]"
    exit 0
fi
IFS=$SavedIFS

if [ -n "${FailureUpdate}" ]; then
    /bin/echo -e "FAILURE-7001: Could not change
 [$PasswordParameter]"\
        "to [$Value] for [$FailureUpdate] account"
    if [ -n "${SuccessUpdate}" ]; then
        /bin/echo -e "Changed [$PasswordParameter]"\
            "to [$Value] for [$SuccessUpdate] account"
    fi
    exit 7001
else
    /bin/echo -e "SUCCESS-7001: Changed [$PasswordParameter]"\
        "to [$Value] for [$SuccessUpdate] account"
    exit 0
fi

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0006755
# AR_TEST_NAME = Verify PASS_MIN_DAYS Setting for Non-system
 Accounts
```

Post Remediation Category	*None*
Remediated Elements	/etc/shadow /etc/shadow-
Post Remediation Steps	No additional Post Remediation steps

2.2.4.60 Verify PASS_WARN_AGE Parameter in /etc/login.defs

Verify PASS_WARN_AGE Parameter in /etc/login.defs

<table>
<tr><td><b>Description</b></td><td>This test verifies that /etc/login.defs is configured to send users warnings at least 7 days before passwords expire.
This setting is used for the creation of new accounts.</td></tr>
<tr><td><b>Severity</b></td><td>0</td></tr>
<tr><td><b>Weight</b></td><td>5</td></tr>
<tr><td><b>Type</b></td><td>Content Test</td></tr>
<tr><td><b>Rules</b></td><td>System Configuration Files</td></tr>
<tr><td><b>Element</b></td><td>Equals "/etc/login.defs"</td></tr>
<tr><td><b>Version conditions</b></td><td>If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*PASS_WARN_AGE[\ \t]+(\d+)[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode)
PASS_WARN_AGE Greater than or equal 7</td></tr>
<tr><td><b>Remediation</b></td><td>To remediate failure of this policy test, set the PASS_WARN_AGE parameter to define the number of days warning given before a password expires.
<b>Setting the PASS_WARN_AGE parameter to define the number of days warning given before a password expires</b>:
1. Become superuser or assume an equivalent role.
2. Open the <b>/etc/login.defs</b> file.
3. Find the line <b>PASS_WARN_AGE <value&gt;</b>.
4. Set the <b>&lt;value&gt;</b> to <b>7</b> or greater and save the file.
For further details, please run the command <b>man login.defs</b> to read man page.</td></tr>
<tr><td><b>Command Line</b></td><td><code>/bin/sh $(ScriptFile.sh)</code></td></tr>
</table>

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/login.defs"
ParameterName="PASS_WARN_AGE"
SeparateSymbol=" "
Value="14"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0013769
# AR_TEST_NAME = Verify PASS_WARN_AGE Parameter in /etc/
login.defs
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

2.2.4.61 Verify PASS_WARN_AGE Setting for Non-system Accounts

Verify PASS_WARN_AGE Setting for Non-system Accounts

Description

This test verifies that all non-system accounts are configured to begin receiving warnings at least 7 days before passwords expire.

Severity

0

Weight

5

Type

Content Test

Rules

Verify Expiration Password for Non-system Account

Element

Equals "Expiration Password"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^\S+:.*PASS_WARN_AGE=[\ \t]*(?:|-\d+|0*[1-6]?|)[\ \t]+.*/ (Flags:Multiline,Comments mode)
'Fail Warning Password Age' for Non-system Accounts Does not exist

Remediation

To remediate failure of this policy test, set the number of days warning given before a password expires to at least 7 for the non-system accounts.

Setting the number of days warning given before a password expires to at least 7 for the non-system accounts:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   for Acc in `awk -F: '$1 !~ /^[[:space:]]*#/ && $3>=500 && $3!
   =65534 {print $1}' /etc/passwd 2>/dev/null`; do awk -F: '$1
   ~ /^[[:space:]]*'$Acc'$/ && $2!~/[!*]+/ && ($6 < 7 || $6 ~ /^[[:s
   pace:]]*$/) {print $1":PASS_WARN_AGE="$6}' /etc/shadow
   2>/dev/null; done
   ```

   to list non-system accounts of which the number of days warning given before a password expires is less than **7**.
3. Change the number of days warning given before a password expires to at least **7** for non-system accounts found in step 2 using the **chage -W <value> <user_name>** command, where **<value>** is greater than or equal to **7**.

For further details, please run the command **man chage** to read man page.

Command Line

`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Option="PASS_WARN_AGE"
Value="14"

FailedAccounts=`/bin/awk -F":" '$1 !~ /[[:space:]]*#/ && $2!~/[!
*]+/ {
    GetIdCmd="/usr/bin/id -u " $1 " 2>/dev/null"; Uid=""
    GetIdCmd | getline Uid
    if(Uid ~ /^[0-9]+$/ && 0+Uid >= 500 && 0+Uid < 65534){
        if($6 !~ /^-?[0-9]+$/ || 0+$6 < 14){ print $1 }
    }
}' /etc/shadow 2>/dev/null`

# Issue the command to change minimum number of days between
 password changes
SavedIFS=$IFS
IFS=`/bin/echo -ne "\n\b"`
for Account in $FailedAccounts; do
    UpdateLog=`/usr/bin/chage -W $Value "$Account" 2>&1`
    if [ -n "$UpdateLog" ]; then
        FailureAccounts=$Account"\n\t"$FailureAccounts
    else
        SuccessAccounts=$Account"\n\t"$SuccessAccounts
    fi
done

if [ -n "$FailureAccounts" ]; then
    FailureAccounts=`/bin/echo -e "$FailureAccounts" | /bin/sed
 '$d'`
    FinalMessage="Could not change value of [$Option] to [$Value]
 for the "
    FinalMessage=$FinalMessage"following account:\n
\t[$FailureAccounts]\n"
fi
IFS=$SavedIFS

if [ -n "$FinalMessage" ]; then
    FinalMessage="FAILURE-7001: "$FinalMessage
    ExitCode=7001
else
    FinalMessage="SUCCESS-7001: "
    ExitCode=0
fi

if [ -n "$SuccessAccounts" ]; then
    SuccessAccounts=`/bin/echo -e "$SuccessAccounts" | /bin/sed
 '$d'`
    FinalMessage=$FinalMessage"Value of [$Option] changed to
 [$Value]"
    FinalMessage=$FinalMessage" for the following account:\n
\t[$SuccessAccounts]"
else
    FinalMessage=`/bin/echo -e "$FinalMessage" | /bin/sed '$d'`
fi

/bin/echo -e "$FinalMessage"
exit $ExitCode


# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0014008
# AR_TEST_NAME = Verify PASS_WARN_AGE Setting for Non-system
 Accounts
```

Post Remediation Category	*None*
Remediated Elements	/etc/shadow /etc/shadow-
Post Remediation Steps	No additional Post Remediation steps

2.2.4.62 Verify That All Groups Defined In The /etc/passwd File Are Defined In The /etc/group File

Verify That All Groups Defined in the /etc/passwd File Are Defined in the /etc/group File

Description

Over time, system administration errors and changes can lead to groups being defined in /etc/passwd but not in /etc/group.
Groups defined in the /etc/passwd file but not in the /etc/group file pose a threat to system security since group permissions are not properly managed.

Severity

0

Weight

5

Type

Content Test

Rules

Get Undefined Groups

Element

Equals "Undefined Groups"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Multiline,Case insensitive,Comments mode)
Undefined Groups Does not exist

Remediation

To remediate failure of this policy test, add undefined group to the /etc/group file.

Adding undefined group to the /etc/group file:

1. Become superuser or assume an equivalent role.
2. Run the script:

```
/bin/awk -F: '$0 !~ /^[[:space:]]*(#|$|(root|bin|daemon|adm|lp|sync|shutdown|halt|mail|news|uucp|operator|games|gopher|ftp|nobody|nscd|vcsa|rpc|mailnull|smmsp|pcap|ntp|dbus|avahi|sshd|rpcuser|nfsnobody|haldaemon|avahi-autoipd|distcache|apache|oprofile|webalizer|dovecot|squid|named|xfs|gdm|sabayon):)/  {print $1, $4}' /etc/passwd 2>/dev/null | while read User Group; do isDefined=`/bin/egrep "^[^:]+:[^:]*:$Group:" /etc/group 2>/dev/null | /bin/egrep -v "^[[:space:]]*(#|:)"`; if [ -z "$isDefined" -o -z "$Group" ]; then /bin/echo "$User:$Group"; fi; done
```

   to list undefined groups.
3. Run the **groupadd -g <gid> <group_name>** command to create undefined groups found in step 2, where **<gid>** is gid of undefined groups found in step 2, **<group_name>** is an optional name.

For further details, please run the command **man groupadd** to read man page.

2.2.4.63 Find All Unowned Directories and Files

Find All Unowned Directories and Files

Description

This test checks for the presence of unowned directories and files on the file system. Any unowned directories and files found on the file system should be carefully reviewed by the system administrator. Unowned directories and files may be an indication of unauthorized system access or improper package maintenance/installation.

Severity

0

Weight

5

Type

Content Test

Rules

Find All Unowned Files

Element

Equals "Unowned Files"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Unowned Files Does not exist

Remediation

To remediate failure of this policy test, set appropriate ownership on the directories and unowned files.

Setting appropriate ownership on the directories and unowned files:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   PARTs=`/bin/df --local -P 2>/dev/null | /bin/egrep -v "/dev/
   sr0" | /bin/awk 'NR != 1  {$1="";$2="";$3="";$4="";$5="";gsub
   ("^[[:space:]]+/","/",$0);print $0}' 2>/dev/null`; SaveIFS=$IFS;
   IFS=`/bin/echo -e "\n\b"`; for PART in $PARTs; do /usr/bin/
   find "$PART" -xdev \( -nouser -o -nogroup \) -print 2>/dev/
   null; done; IFS=$SaveIFS
   ```

   to list all directories and unowned files.
3. Check the ownership of the above directories and files using the **/bin/ls -ldL <file_location>** command.
4. Change ownership using the **/bin/chown <user_owner>:<group_owner> <file_location>** command if needed.

2.2.5 Remove All Unnecessary Functionality

Remove all unnecessary functionality, such as scripts, drivers, features, subsystems, file systems, and unnecessary web servers.

2.2.5.1 Verify That the xorg-x11-server-common (X Windows) Package Is Not Installed

Verify That the xorg-x11-server-common (X Windows) Package Is Not Installed

Description	This test verifies that the xorg-x11-server-common (X Windows) package must not be installed. Removing all packages which constitute the X Window System ensures users or malicious software cannot start X.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*xorg-x11-server-common-\d.*$/ (Flags:Multiline,Comments mode) X Windows Package Does not exist
Remediation	To remediate failure of this policy test, erase the X Windows package. **Erasing the X Windows package:** 1. Become superuser or assume an equivalent role. 2. Run the **yum erase xorg-x11-server-common** command to remove **X Windows** package. For further details, please run the command **man yum** to read man page.

2.2.5.2 Verify That SSH X11 Forwarding Is Disabled

Verify That SSH X11 Forwarding Is Disabled

Description	The X11Forwarding parameter provides the ability to tunnel X11 traffic through the connection to enable remote graphic connections. Disable X11 forwarding unless there is an operational requirement to use X11 applications directly. There is a small risk that the remote X11 servers of users who are logged in via SSH with X11 forwarding could be compromised by other users on the X11 server. Note that even if X11 forwarding is disabled, users can always install their own forwarders.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*X11Forwarding[\ \t]+(\S+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) (SSH X11 Forwarding Equals "no" AND SSH X11 Forwarding Setting Exists) OR SSH X11 Forwarding Setting Does not exist
Remediation	To remediate failure of this policy test, configure the SSH server to disable X11 Forwarding. **Configuring the SSH server to disable X11 Forwarding:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **X11Forwarding <value>**. 4. Set **<value>** to **no** and save the file. 5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.

2.2.5.3 Verify That Ctrl-Alt-Delete Sequence Is Disabled

Verify That Ctrl-Alt-Delete Sequence Is Disabled

Description	This test verifies that Linux systems have disabled the <CTRL><ALT><DELETE> key sequence. A locally logged-in user who presses Ctrl-Alt-Del, when at the console, can reboot the system. If accidentally pressed, as could happen in the case of mixed OS environment, this can create the risk of short-term loss of availability of systems due to unintentional reboot. In the GNOME graphical environment, risk of unintentional reboot from the Ctrl-Alt-Del sequence is reduced because the user will be prompted before any action is taken.
Severity	0
Weight	5
Type	Content Test
Rules	Disable Ctrl-Alt-Delete Key Sequence
Element	Equals "Disable Ctrl-Alt-Delete Key Sequence"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*exec[\ \t]+(?:/sbin/)?shutdown[\ \t]+.*$/ (Flags:Multiline,Comments mode) Ctrl-Alt-Delete Sequence Does not exist
Remediation	To remediate failure of this policy test, disable Ctrl-Alt-Delete sequence. **Disabling Ctrl-Alt-Delete sequence:** 1. Become superuser or assume an equivalent role. 2. Run the **/bin/grep -l control-alt-delete /etc/init/*** command to list all control-alt-delete files configuration. 3. Open each file listed in the step 2. 4. Find the line that contains **exec /sbin/shutdown** entry, such as **exec /sbin/shutdown -r now "Control-Alt-Delete pressed"**. 5. Comment out the line and save the file.

2.2.5.4 Verify That Unconfined Daemons Are Disabled

Verify That Unconfined Daemons Are Disabled

Description

Daemons that are not defined in SELinux policy will inherit the security context of their parent process. Since daemons are launched and descend from the init process, they will inherit the security context label initrc_t. This could cause the unintended consequence of giving the process more permission than it requires.

Severity

0

Weight

5

Type

Content Test

Rules

Get Unconfined Daemons

Element

Equals "Get Unconfined Daemons"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Unconfined Daemons Does not exist

Remediation

To remediate failure of this policy test, check for unconfined daemons.

Checking for unconfined daemons:

1. Perform the following to determine if unconfined daemons are running on the system:

   ```
   ps -eZ | egrep "initrc" | egrep -vw "tr|ps|egrep|bash|awk"
   ```

2. Investigate any unconfined daemons found in step 1.
3. Using the following command to kill daemon process that you want to kill:

   ```
   kill -9 <PID>
   ```

   <PID> that is PID of process that list in second column in step 1.

2.2.5.5 Verify That X Windows Is Not Installed on the System

Verify That X Windows Is Not Installed on the System

Description	The X Windows system provides a Graphical User Interface (GUI) where users can have multiple windows in which to run programs and various add on. The X Windows system is typically used on desktops where users login, but not on servers where users typically do not login. Unless your organization specifically requires graphical login access via the X Windows System, remove the server to reduce the potential attack surface.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*xorg-x11-server-.*$/ (Flags:Multiline,Comments mode) X Window System Does not exist
Remediation	To remediate failure of this policy test, remove software group "X Window System" **Remove software group "X Window System":** 1. Become superuser or assume an equivalent role. 2. Run the **rpm -qa | grep xorg-x11** command to list all Xorg packages. 3. Remove all packages listed in step 2.

2.2.5.6 Verify That GUI Login Is Disabled

Verify That GUI Login Is Disabled

Description

This test checks that the GUI login is disabled.
Systems configured for GUI login run at run-level 5. Disabling the GUI login causes the system to run at run-level 3, which is more desirable than running at run-level 5.
NOTE: A user may still run X Windows from run-level 3 by typing 'startx'.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/inittab"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^\s*id:5:initdefault:\s*(?:$|\#)/ (Flags:Multiline,Comments mode)
GUI Login Setting Does not exist

Remediation

To remediate failure of this policy test, configure the /etc/inittab file to disable GUI login.

Configuring the /etc/inittab file to disable GUI login:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/inittab** file.
3. Find the line **id:<number>:initdefault:**.
4. Set the **<number>** such that it is not equal to **5** and save the file.
   id:3:initdefault:

For further details, please run the command **man inittab** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/inittab"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to disable GUI Login
UpdateLog=`(/bin/awk -F":"\
    '$0 ~ /^[[:space:]]*id:5:initdefault:[[:space:]]*$/\
    {$0="id:3:initdefault:"}; {print $0}' $BackupName >
 $FileName) 2>&1`
if [ -n "$UpdateLog" ]; then
    /bin/echo "FAILURE-7001: Could not update the [$FileName]"\
        "to disable GUI Login"
 # Rollback to the original file
    /bin/cp -f $BackupName $FileName 2>/dev/null
    exit 7001
else
    /bin/echo "SUCCESS-7001: Updated the [$FileName] file to"\
        "disable GUI Login successfully"
    exit 0
fi


# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0000945
# AR_TEST_NAME = Verify That GUI Login Is Disabled
```

Post Remediation Category

None

Remediated Elements

`/etc/inittab`

Post Remediation Steps	No additional Post Remediation steps

2.2.5.7 Verify That Hard Core-dumps Are Disabled

Verify That Hard Core-dumps Are Disabled

Description

This test determines whether hard core-dump limits have been set to zero in /etc/security/limits.conf. This setting supports information confidentiality by preventing potentially sensitive information from being leaked to a core file on a hardware failure.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/security/limits.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]**[\ \t]+hard[\ \t]+core[\ \t]+0[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode)
Hard Core-dumps Setting Exists

Remediation

To remediate failure of this policy test, set hard core to disable core dumps in order to prevent the destruction of large amounts of disk space that may contain sensitive data.

Setting hard core to disable core dumps:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/security/limits.conf** file.
3. Find the lines * **hard core <value>** or add it to file (if not found).
4. Set the **<value>** to **0** and save the file.

For further details, please run the command **man limits.conf** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/security/limits.conf"
ParameterName="*\t\thard\tcore\t\t"
Regex="\*[[:space:]]+hard[[:space:]]+core"
SeparateSymbol=" "
Value="0"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo -e "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo -e "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$0 ~ \
    /^[[:space:]]*'"$Regex"'[[:space:]]+/ {print}' "$FileName"
 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '($0 ~ /^[[:space:]]*'"$Regex"'[[:space:]]/) {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo -e "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo -e "SUCCESS-4001: Value of [$ParameterName]
 parameter changed to" \
        "[$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo -e
 "${ParameterName}${SeparateSymbol}${Value}" >> \
        "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo -e "FAILURE-4002: Could not add" \
            "[${ParameterName}${SeparateSymbol}${Value}]
 parameter to" \
                "[$FileName] file"
        exit 4002
    fi
    /bin/echo -e "SUCCESS-4002:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "parameter added to [$FileName] file"
fi
exit 0


# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0013863
# AR_TEST_NAME = Verify That Hard Core-dumps Are Disabled
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

2.3 Encrypt Non-console Administrative Access

Encrypt all non-console administrative access using strong cryptography. Use technologies such as SSH, VPN, or SSL/TLS for web-based management and other non-console administrative access.

2.3.1 Verify That the Algorithm Used for Password Hashing Is SHA-512 (login.defs)

Verify That the Algorithm Used for Password Hashing Is SHA-512 (login.defs)

Description	This test verifies that the algorithm use for password hashing is SHA-512 (login.defs). Systems must employ cryptographic hashes for passwords using the SHA-512 algorithm. The use of unapproved algorithms may result in weak password hashes more vulnerable to compromise.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/login.defs"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*ENCRYPT_METHOD[\ \t]+(SHA\d+)[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode) Password Hashing Algorithm Equals "SHA512"
Remediation	To remediate failure of this policy test, set the password hashing algorithm to sha512. **Setting the password hashing algorithm to sha512**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/login.defs** file. 3. Find the line **ENCRYPT_METHOD <value>**. 4. Set the **<value>** to **SHA512** save the file. For further details, please run the command **man login.defs** to read man page.

2.3.2 Verify the Algorithm Used for Password Hashing Is SHA-512 (/etc/libuser.conf)

Verify the Algorithm Used for Password Hashing Is SHA-512 (/etc/libuser.conf)

Description

This test verifies that the algorithm use for password hashing is SHA-512 (/etc/libuser.conf).
Systems must employ cryptographic hashes for passwords using the SHA-512 algorithm. The use of unapproved algorithms may result in weak password hashes more vulnerable to compromise.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/libuser.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*\[defaults\][\ \t]*(?:$|\#)\n(?:^(?![\ \t]*\[).*$\n)*^[\ \t]*crypt_style[\ \t]*=[\ \t]*(?i:sha512)[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode)
SHA-512 Algorithm Exists

Remediation

To remediate failure of this policy test, set the password hashing algorithm to sha512.

Setting the password hashing algorithm to sha512:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/libuser.conf** file.
3. Find the line **crypt_style = <value>** in the **[default]** section.
4. Set the **<value>** to **sha512** and save the file.

For further details, please run the command **man libuser.conf** to read man page.

2.3.3 Verify That the Algorithm Used for Password Hashing Is SHA-512 (system-auth)

Verify That the Algorithm Used for Password Hashing Is SHA-512 (system-auth)

Description

This test verifies that the algorithm use for password hashing is SHA-512.
Systems must employ cryptographic hashes for passwords using the SHA-512 algorithm. The use of unapproved algorithms may result in weak password hashes more vulnerable to compromise.

Severity

0

Weight

5

Type

Content Test

Rules

Get All Configurations in system-auth File

Element

Equals "Get All Configurations in system-auth File"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bpam_unix\.so(?:[\ \t]+[^\#&&\S]+)*[\ \t]+sha512[\ \t]+.*$/ (Flags:Multiline,Case insensitive,Comments mode)
SHA-2 of Algorithms Exists

Remediation

To remediate failure of this policy test, use SHA-512 as default password hashing algorithm.

Using SHA-512 as default password hashing algorithm:

1. Become superuser or assume an equivalent role.
2. Run following command to to list the paths of the PAM configuration files need to update:

   ```
   PamConfigFile="/etc/pam.d/system-auth"; directory="/etc/pam.d"; GetIncludeConfig(){ Files="";FileConfig=$1;ModuleInterface=$2;IncludeFiles=`/bin/cat "$FileConfig" 2>/dev/null | /bin/awk -F"#" ' $1 ~ /^[[:space:]]*('$ModuleInterface')[[:space:]]+(required|requisite)[[:space:]]+(.*\/)?pam_stack\.so[[:space:]]+service=\w+/ {print $1}' | /bin/awk -F"service=" '{print $2}' | /bin/awk 'BEGIN {ORS=";"} {print $1}'`;Files=$Files$IncludeFiles; IncludeFiles=`/bin/cat "$FileConfig" 2>/dev/null | /bin/awk -F"#" ' $0 ~ /^[[:space:]]*('$ModuleInterface')[[:space:]]+include[[:space:]].*/ {print $1}' | /bin/awk 'BEGIN {ORS=";"} {print $3}'`;  Files=$Files$IncludeFiles; SaveIFS=$IFS; IFS=";";for file in $Files; do if [ "`/usr/bin/dirname $file 2>/dev/null`" != "." ]; then if [ -e "$file" ]; then /bin/echo $file 2>/dev/null ;  GetIncludeConfig  "$file" "$ModuleInterface";fi; else full_path=$directory"/"$file; if [ -e "$full_path" ]; then /bin/echo $full_path ;GetIncludeConfig  "$full_path" "$ModuleInterface";    fi; fi; done;IFS=$SaveIFS; } ;/bin/echo "$PamConfigFile" 2>/dev/null; GetIncludeConfig "$PamConfigFile" "auth";GetIncludeConfig "$PamConfigFile" "password"; GetIncludeConfig "$PamConfigFile" "session"; GetIncludeConfig "$PamConfigFile" "account";
   ```

3. For each file listed in step 2, find the line that contains:

   password <control_flag> *[security_path/]*pam_unix.so *[parameters]*

4. If the line is found, add the hash algorithm **sha512** such as :

   password sufficient *[security_path/]* pam_unix.so sha512 *[parameters]*

5. If the line is not found in any file listed in step 2, review the **/etc/pam.d/system-auth** file, then edit or add some entries if needed to make sure that the file contains the following ordered lines:

   password sufficient *[security_path/]*pam_unix.so sha512 *[parameters]*
   password required *[security_path/]*pam_deny.so

6. Save the file.

For further details, please run the **man pam_unix** command to read man page.

2.3.4 Verify That Password Hashing Algorithm Is Upgraded to SHA-512

Verify That Password Hashing Algorithm Is Upgraded to SHA-512

Description	The SHA-512 encryption has been available since Red Hat release 5.2,. The commands below change password encryption from md5 to sha512 (a much stronger hashing algorithm). All existing accounts will need to perform a password change to upgrade the stored hashes to the new algorithm.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/sysconfig/authconfig"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^PASSWDALGORITHM=sha512[\ \t]*$/ (Flags:Multiline,Comments mode) PASSWDALGORITHM Setting Exists
Remediation	To remediate failure of this policy test, upgrading password hashing algorithm to SHA-512. **Upgrading password hashing algorithm to SHA-512**: 1. Become superuser or assume an equivalent role. 2. Run the **authconfig --passalgo=sha512 --update** command to upgrade password hashing algorithm to **SHA-512**. 3. Run the following command to force users to change their passwords on next login: **awk -F: '(0+$3 >=500 && $1 != "nfsnobody") { print $1 }' /etc/passwd | xargs -n 1 chage -d 0** For further details, please run the command **man authconfig** to read man page.

2.3.5 Verify That SSH Uses Approved Ciphers during Communication

Verify That SSH Uses Approved Ciphers during Communication

Description

This variable limits the types of ciphers that SSH can use during communication.
Based on research conducted at various institutions, it was determined that the symmetric portion of the SSH Transport Protocol (as described in RFC 4253) has security weaknesses that allowed recovery of up to 32 bits of plaintext from a block of ciphertext that was encrypted with the Cipher Block Chaining (CBC) method. From that research, new Counter mode algorithms (as described in RFC4344) were designed that are not vulnerable to these types of attacks and these algorithms are now recommended for standard use.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/ssh/sshd_config"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*(?i)Ciphers(?-i)[\ \t]*((?:aes128-ctr|aes192-ctr|aes256-ctr)\b,?)+[\ \t]*(?:$|\#.*)$/ (Flags:Multiline,Comments mode)
Approved Ciphers Configuration Exists

Remediation

To remediate failure of this policy test, configure the SSH server to specify the ciphers allowed for protocol version 2.

Configuring the SSH server to specify the ciphers allowed for protocol version 2:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Find the line **Ciphers <value>**.
4. Set **<value>** where **<value>** does not contain ciphers which are different from **aes128-ctr**, **aes192-ctr**, **aes256-ctr** and save the file.
5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

For further details, please run the command **man sshd_config** to read man page.

2.3.6 TFTP Secure Mode

TFTP Secure Mode

Description	This test verifies that the secure mode option is used if TFTP is implemented on a system that supports it. TFTP that is used without the secure mode option sends data out in clear text which can easily be intercepted and read.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/xinetd.d/tftp"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*server_args[\ \t]*=[\ \t]*([^\#\n]+[\ \t])?-s\b.*/ (Flags:Multiline,Comments mode) TFTP Secure Mode Setting Exists
Remediation	To remediate failure of this policy test, add "-s" parameter to ensure that TFTP is run in secure mode. **Adding "-s" parameter to ensure that TFTP is run in secure mode**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/xinetd.d/tftp** file. 3. Find the line that contains **server_args**. 4. Add "**-s**" to the line as **server_args = -s <tftp_boot>**. 5. Run the **/etc/init.d/xinetd restart** command to apply the changes. For further details, please refer to: http://www.redhat.com/mirrors/LDP/HOWTO/Clone-HOWTO/setting-up.html

2.3.7 Verify That sshd_config Uses Protocol 2 Only

Verify That sshd_config Uses Protocol 2 Only

Description	This test verifies that the SSH server uses SSH version 2 only. SSH version 1 contains a number of security vulnerabilities. SSH version 2 addresses these vulnerabilities and should be used instead of SSH version 1.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*Protocol[\ \t]+(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) SSH Server Protocol Version Equals 2
Remediation	To remediate failure of this policy test, configure the SSH daemon to use safe defaults for the server by setting the Protocol 2. **Configuring the SSH Server to set the Protocol 2**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **Protocol <value>**. 4. If found, then set **<value>** to **2** and save the file. 5. If not found, then add the **Protocol 2** line to the file and save it. 6. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
ParameterName="Protocol"
SeparateSymbol=" "
Value="2"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
                "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '{IGNORECASE=1;} $1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}'
 ${FileName} \
    2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '{IGNORECASE=1;} ($1 ~ /
^[[:space:]]*'"$ParameterName"'[[:space:]]*$/) \
        {$0 = Line;}{print}'
 Line="${ParameterName}${SeparateSymbol}${Value}" \
        ${BackupName} > ${FileName}) 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]"\
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName]"\
        "parameter changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> $FileName) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line
 to"\
            "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]"\
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_CASE_INSENSITIVE
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0003256
# AR_TEST_NAME = Verify That sshd_config Uses Protocol 2 Only


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>pkill -HUP sshd</b> or <b>/
sbin/service sshd restart</b> commands to restart the <b>sshd </
b>service.</li></ol>
```

Post Remediation Category `Other`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

Requirement 4 Encrypt Transmission of Cardholder Data across Open, Public Networks

Sensitive information must be encrypted during transmission over networks that are easily accessed by malicious individuals. Misconfigured wireless networks and vulnerabilities in legacy encryption and authentication protocols continue to be targets of malicious individuals who exploit these vulnerabilities to gain privileged access to cardholder data environments.

4.1 Use Strong Cryptography and Security Protocols

Use strong cryptography and security protocols (for example, SSL/TLS, IPSEC, SSH, etc.) to safeguard sensitive cardholder data during transmission over open, public networks, including the following:
- Only trusted keys and certificates are accepted.
- The protocol in use only supports secure versions or configurations.
- The encryption strength is appropriate for the encryption methodology in use.
Examples of open, public networks include but are not limited to:
- The Internet
- Wireless technologies, including
802.11 and Bluetooth
- Cellular technologies, for example, Global System for Mobile communications (GSM), Code division multiple access (CDMA)
- General Packet Radio Service (GPRS).
- Satellite communications.

4.1.0 Use Strong Cryptography and Security Protocols Over Non-wireless Networks

Use strong cryptography and security protocols (for example, SSL/TLS, IPSEC, SSH, etc.) to safeguard sensitive cardholder data during transmission over open, public networks, including the following:
- Only trusted keys and certificates are accepted.
- The protocol in use only supports secure versions or configurations.
- The encryption strength is appropriate for the encryption methodology in use.
Examples of open, public networks include but are not limited to:
- The Internet
- Wireless technologies, including
802.11 and Bluetooth
- Cellular technologies, for example, Global System for Mobile communications (GSM), Code division multiple access (CDMA)
- General Packet Radio Service (GPRS).
- Satellite communications.

4.1.0.1 Verify That LDAP Is Configured to Use TLS for All Transactions

Verify That LDAP Is Configured to Use TLS for All Transactions

Description	The ssl directive specifies whether to use ssl or not. If not specified it will default to no. It should be set to start_tls rather than doing LDAP over SSL.
Severity	0
Weight	5
Type	Content Test
Rules	Get Content of pam_ldap.conf File
Element	Equals "Get Content of pam_ldap.conf File"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*ssl[\ \t]+start_tls[\ \t]*$/ (Flags:Multiline,Comments mode) TLS Configuration Exists

Remediation

To remediate failure of this policy test, configure LDAP to use TLS for all transactions.

To configure LDAP to use TLS for all transactions:

1. Become superuser or assume an equivalent role.
2. Run the following command to check **pam_ldap** package is installed or not:

   ```
   PamLDAPPackage=`/bin/rpm -qa 2>/dev/null | /bin/egrep
   "pam_ldap"`; if [ -z "$PamLDAPPackage" ]; then /bin/echo
   "pam_ldap package is not installed";fi
   ```
3. To install **pam_ldap** package run the following command:

   ```
   rpm -ivh <pam_ldap package>
   ```
4. Open **/etc/pam_ldap.conf** file.
5. Edit this file to contain the following values:

   ```
   base dc=example,dc=org
   uri ldap://server.example.org/
   ssl start_tls
   tls_cacert /path/to/your/cacert.crt
   ```

Note: If the system does not use LDAP for authentication or account information, this test can be skipped

For further details, please refer to :

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/ch-Directory_Servers.html

4.1.0.2 Verify That TLS Is Configured with Trust Certificates

Verify That TLS Is Configured with Trust Certificates

Description

The tls_cacertdir or tls_cacertfile directives are required when tls_checkpeer is configured (which is the default for openldap versions 2.1 and up). These directives define the path to the trust certificates signed by the site CA.

Severity

0

Weight

5

Type

Content Test

Rules

Get Content of pam_ldap.conf File

Element

Equals "Get Content of pam_ldap.conf File"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*(?:tls_cacertdir|tls_cacertfile)[\ \t]+\S+.*$/ (Flags:Multiline,Comments mode)
Trust Certificates Exists

Remediation

To remediate failure of this policy test, configure LDAP to use TLS using trust certificates signed by the site CA.

To configure LDAP to use TLS using trust certificates signed by the site CA:

1. Become superuser or assume an equivalent role.
2. Run the following command to check **pam_ldap** package is installed or not:

```
PamLDAPPackage=`/bin/rpm -qa 2>/dev/null | /bin/egrep
"pam_ldap"`; if [ -z "$PamLDAPPackage" ]; then /bin/echo
"pam_ldap package is not installed";fi
```

3. To install **pam_ldap** package run the following command:

```
rpm -ivh <pam_ldap package>
```

4. Open **/etc/pam_ldap.conf** file.
5. Ensure that the file has the following line:

```
tls_cacertdir <cert directory>
or
tls_cacertfile <cert file>
```

For further details, please refer to :

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/ch-Directory_Servers.html

4.1.0.3 Verify That Password Hashing Algorithm Is Upgraded to SHA-512

Verify That Password Hashing Algorithm Is Upgraded to SHA-512

Description	The SHA-512 encryption has been available since Red Hat release 5.2,. The commands below change password encryption from md5 to sha512 (a much stronger hashing algorithm). All existing accounts will need to perform a password change to upgrade the stored hashes to the new algorithm.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/sysconfig/authconfig"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^PASSWDALGORITHM=sha512[\ \t]*$/ (Flags:Multiline,Comments mode) PASSWDALGORITHM Setting Exists
Remediation	To remediate failure of this policy test, upgrading password hashing algorithm to SHA-512. **Upgrading password hashing algorithm to SHA-512**: 1. Become superuser or assume an equivalent role. 2. Run the **authconfig --passalgo=sha512 --update** command to upgrade password hashing algorithm to **SHA-512**. 3. Run the following command to force users to change their passwords on next login: **awk -F: '(0+$3 >=500 && $1 != "nfsnobody") { print $1 }' /etc/passwd | xargs -n 1 chage -d 0** For further details, please run the command **man authconfig** to read man page.

4.1.0.4 Verify That sshd_config Uses Protocol 2 Only

Verify That sshd_config Uses Protocol 2 Only

Description	This test verifies that the SSH server uses SSH version 2 only. SSH version 1 contains a number of security vulnerabilities. SSH version 2 addresses these vulnerabilities and should be used instead of SSH version 1.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*Protocol[\ \t]+(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) SSH Server Protocol Version Equals 2
Remediation	To remediate failure of this policy test, configure the SSH daemon to use safe defaults for the server by setting the Protocol 2. **Configuring the SSH Server to set the Protocol 2**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **Protocol <value>**. 4. If found, then set **<value>** to **2** and save the file. 5. If not found, then add the **Protocol 2** line to the file and save it. 6. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
ParameterName="Protocol"
SeparateSymbol=" "
Value="2"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
                "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '{IGNORECASE=1;} $1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}'
 ${FileName} \
    2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '{IGNORECASE=1;} ($1 ~ /
^[[:space:]]*'"$ParameterName"'[[:space:]]*$/) \
        {$0 = Line;}{print}'
 Line="${ParameterName}${SeparateSymbol}${Value}" \
        ${BackupName} > ${FileName}) 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]"\
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName]"\
        "parameter changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> $FileName) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line
 to"\
            "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]"\
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_CASE_INSENSITIVE
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0003256
# AR_TEST_NAME = Verify That sshd_config Uses Protocol 2 Only


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>pkill -HUP sshd</b> or <b>/
sbin/service sshd restart</b> commands to restart the <b>sshd </
b>service.</li></ol>
```

Post Remediation Category Other

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

4.1.0.5 Verify That SSH Uses Approved Ciphers during Communication

Verify That SSH Uses Approved Ciphers during Communication

Description

This variable limits the types of ciphers that SSH can use during communication.
Based on research conducted at various institutions, it was determined that the symmetric portion of the SSH Transport Protocol (as described in RFC 4253) has security weaknesses that allowed recovery of up to 32 bits of plaintext from a block of ciphertext that was encrypted with the Cipher Block Chaining (CBC) method. From that research, new Counter mode algorithms (as described in RFC4344) were designed that are not vulnerable to these types of attacks and these algorithms are now recommended for standard use.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/ssh/sshd_config"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*(?i)Ciphers(?-i)[\ \t]*((?:aes128-ctr|aes192-ctr|aes256-ctr)\b,?)+[\ \t]*(?:$|\#.*)$/ (Flags:Multiline,Comments mode)
Approved Ciphers Configuration Exists

Remediation

To remediate failure of this policy test, configure the SSH server to specify the ciphers allowed for protocol version 2.

Configuring the SSH server to specify the ciphers allowed for protocol version 2:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Find the line **Ciphers <value>**.
4. Set **<value>** where **<value>** does not contain ciphers which are different from **aes128-ctr**, **aes192-ctr**, **aes256-ctr** and save the file.
5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

For further details, please run the command **man sshd_config** to read man page.

Requirement 7 Restrict Access to Cardholder Data by Business Need to Know

To ensure critical data can only be accessed by authorized personnel, systems and processes must be in place to limit access based on need to know and according to job responsibilities.
"Need to know" is when access rights are granted to only the least amount of data and privileges needed to perform a job.

7.1 Access Restrictions

Limit access to system components and cardholder data to only those individuals whose job requires such access.

7.1.2 Enforce Least Privilege

Restriction of access rights to privileged user IDs to least privileges necessary to perform job responsibilities.

7.1.2. 1 Verify That 'nodev' Option Is Added to /tmp Partition in the /etc/fstab File

Verify That 'nodev' Option Is Added to /tmp Partition in the /etc/fstab File

Description	The nodev mount option specifies that the filesystem cannot contain special devices. Since the /tmp filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/tmp[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bnodev\b.*$/ (Flags:Multiline,Comments mode) /tmp with nodev Option Exists
Remediation	To remediate failure of this policy test, set nodev option for /tmp partition. **Setting nodev option for /tmp partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/tmp**. 4. If not found, use the **Logical Volume Manager (LVM)** to create a separate partition for **/tmp** then go to step 5. 5. If found, add the **nodev** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nodev /tmp** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 2 Verify That 'nosuid' Option Is Added to /tmp Partition in the /etc/fstab File

Verify That 'nosuid' Option Is Added to /tmp Partition in the /etc/fstab File

Description	The nosuid mount option specifies that the filesystem cannot contain set userid files. Since the /tmp filesystem is only intended for temporary file storage, set this option to ensure that users cannot create set userid files in /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/tmp[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bnosuid\b.*$/ (Flags:Multiline,Comments mode) /tmp with nosuid Option Exists
Remediation	To remediate failure of this policy test, set the nosuid option for the /tmp partition. **Setting the nosuid option for the /tmp partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/tmp**. 4. If not found, use the **Logical Volume Manager (LVM)** to create a separate partition for **/tmp** then go to step 5. 5. If found, add the **nosuid** option to the fourth field, using a comma to separate from other options. 6. Remount the partition by using the **mount -o remount,nosuid /tmp** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 3 Verify That /tmp Partition Mounted with 'nosuid'

Verify That /tmp Partition Mounted with 'nosuid'

Description	The nosuid mount option specifies that the filesystem cannot contain set userid files. Since the /tmp filesystem is only intended for temporary file storage, set this option to ensure that users cannot create set userid files in /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/tmp[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnosuid\b.*\).*$/ (Flags:Multiline,Comments mode) /tmp with nosuid Option Exists
Remediation	To remediate failure of this policy test, set nosuid option for /tmp partition. **Setting nosuid option for /tmp partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/tmp**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/tmp**, then go to step 5. 5. If found, add the **nosuid** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nosuid /tmp** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 4 Verify That 'noexec' Option Is Added to /tmp Partition in the /etc/fstab File

Verify That 'noexec' Option Is Added to /tmp Partition in the /etc/fstab File

Description	The noexec mount option specifies that the filesystem cannot contain executable binaries. Since the /tmp filesystem is only intended for temporary file storage, set this option to ensure that users cannot run executable binaries from /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/tmp[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bnoexec\b.*$/ (Flags:Multiline,Comments mode) /tmp with noexec Option Exists
Remediation	To remediate failure of this policy test, set noexec option for /tmp partition. **Set noexec option for /tmp partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/tmp**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/tmp**, then go to step 5. 5. If found, add the **noexec** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,noexec /tmp** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 5 Verify That /tmp Partition Mounted with 'noexec'

Verify That /tmp Partition Mounted with 'noexec'

Description	The noexec mount option specifies that the filesystem cannot contain executable binaries. Since the /tmp filesystem is only intended for temporary file storage, set this option to ensure that users cannot run executable binaries from /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/tmp[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnoexec\b.*\).*$/ (Flags:Multiline,Comments mode) /tmp with noexec Option Exists
Remediation	To remediate failure of this policy test, set noexec option for /tmp partition. **Set noexec option for /tmp partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/tmp**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/tmp**, then go to step 5. 5. If found, add the **noexec** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,noexec /tmp** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 6 Verify That 'nodev' Option Is Added to /home Partition in the /etc/fstab File

Verify That 'nodev' Option Is Added to /home Partition in the /etc/fstab File

Description	The nodev mount option specifies that the filesystem cannot contain special devices. Since the /home filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /home.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/home[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bnodev\b.*$/ (Flags:Multiline,Comments mode) /home with nodev Option Exists
Remediation	To remediate failure of this policy test, set nodev option for /home partition. **Setting nodev option for /home partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/home**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/home**, then go to step 5. 5. If found, add the **nodev** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nodev /home** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 7 Verify That 'nodev' Option Is Added to /dev/shm Partition in the /etc/fstab File

Verify That 'nodev' Option Is Added to /dev/shm Partition in the /etc/fstab File

Description	The nodev mount option specifies that the filesystem cannot contain special devices. Since the /dev/shm filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /dev/shm.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/dev/shm[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bnodev\b.*$/ (Flags:Multiline,Comments mode) Right Configuration Exists
Remediation	To remediate failure of this policy test, set nodev option for /dev/shm partition. **Setting nodev option for /dev/shm partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/dev/shm**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/dev/shm**, then go to step 5. 5. If found, add the **nodev** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nodev /dev/shm** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 8 Verify That /dev/shm Partition Is Set nosuid Option in /etc/fstab

Verify That /dev/shm Partition Is Set nosuid Option in /etc/fstab

Description

The nosuid mount option specifies that the /dev/shm (temporary filesystem stored in memory) will not execute setuid and setgid on executable programs as such, but rather execute them with the uid and gid of the user executing the program. Setting this option on a file system prevents users from introducing privileged programs onto the system and allowing non-root users to execute them.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/fstab"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/dev/shm[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]*\bnosuid\b.*$/ (Flags:Multiline,Comments mode)
/dev/shm with nosuid Option Exists

Remediation

To remediate failure of this policy test, set nosuid option for /dev/shm partition.

Setting nosuid option for /dev/shm partition:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/fstab** file.
3. Find the line with options for **/dev/shm**.
4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/dev/shm**, then go to step 5.
5. If found, add the **nosuid** option to the fourth field, using a comma to separate from other options.
6. Remount partition by using the **mount -o remount, nosuid /dev/shm** command.

For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems:

http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2. 9 Verify That /dev/shm Partition Is Set noexec Option in /etc/fstab

Verify That /dev/shm Partition Is Set noexec Option in /etc/fstab

Description	Set noexec on the shared memory partition to prevent programs from executing from there. Setting this option on a file system prevents users from executing programs from shared memory. This deters users from introducing potentially malicious software on the system
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/fstab"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+/dev/shm[\ \t]+[^\#&&\S]+[\ \t]+[^\#&&\S]* \bnoexec\b.*$/ (Flags:Multiline,Comments mode) /dev/shm with noexec Option Exists
Remediation	To remediate failure of this policy test, set noexec option for /dev/shm partition. **Setting noexec option for /dev/shm partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/dev/shm**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/dev/shm**, then go to step 5. 5. If found, add the **noexec** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,noexec /dev/shm** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2.10 Verify Ownership on All Files and Directories Associated with the audit Package

Verify Ownership on All Files and Directories Associated with the audit Package

Description	This test verifies that ownership on all files and directories associated with the "audit" package. Ownership of audit binaries and configuration files that is incorrect could allow an unauthorized user to gain privileges that they should not have. The ownership set by the vendor should be maintained. Any deviations from this baseline should be investigated.
Severity	0
Weight	5
Type	Content Test
Rules	Get Permissions on All Files and Directories Associated with the audit Package
Element	Equals "Permissions on All Files and Directories Associated with the audit Package"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^.{5}U.{3}[\ \t]+.*$/ (Flags:Multiline,Comments mode) User Ownership Differs Does not exist
Remediation	To remediate failure of this policy test, restore ownership of audit package files and directories from what is expected by the RPM database. **Restoring ownership of audit package files and directories:** 1. Become superuser or assume an equivalent role. 2. Run the following command: **/bin/rpm --setugids audit** to set user/group ownership of files in the given package.

7.1.2.11 Verify Group-ownership on All Files and Directories Associated with the "audit" Package

Verify Group-ownership on All Files and Directories Associated with the "audit" Package

Description	This test verifies that group-ownership on all files and directories associated with the "audit" package. Group-ownership of audit binaries and configuration files that is incorrect could allow an unauthorized user to gain privileges that they should not have. The group-ownership set by the vendor should be maintained. Any deviations from this baseline should be investigated.
Severity	0
Weight	5
Type	Content Test
Rules	Get Permissions on All Files and Directories Associated with the audit Package
Element	Equals "Permissions on All Files and Directories Associated with the audit Package"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^.{6}G.{2}[\ \t]+.*$/ (Flags:Multiline,Comments mode) Group Ownership Differs Does not exist
Remediation	To remediate failure of this policy test, restore group-ownership of audit package files and directories from what is expected by the RPM database. **Restoring group-ownership of audit package files and directories:** 1. Become superuser or assume an equivalent role. 2. Run the following command: **/bin/rpm --setugids audit** to set user/group ownership of files in the given package.

7.1.2.12 Verify That All System Command Files Have Appropriate Mode

Verify That All System Command Files Have Appropriate Mode

Description

This test verifies that all system command files are owned by 'root' and have permissions set to 755 or more restrictive.
Restricting permissions will protect system command files from unauthorized modification. System command files include files present in directories used by the operating system for storing default system executables and files present in directories included in the system's default executable search paths.

Severity

0

Weight

5

Type

Content Test

Rules

Get System Command Files with Invalid Permissions

Element

Equals "Permissions of All System Command Files"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Multiline,Case insensitive,Comments mode)
Command File Permissions Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions for all system command files.

Setting appropriate permissions for all system command files:

1. Become superuser or assume an equivalent role.
2. Run following command to list all invalid system command files:

```
/usr/bin/find -L /bin /usr/bin /usr/local/bin /sbin /usr/sbin /usr/local/sbin \( -perm -g+w -o -perm -o+w -o ! -user root \) -type f -ls 2>/dev/null | /bin/awk '{print $3, $5, $6, $NF}' 2>/dev/null
```

3. Remove write permission for group and others on all system command files found in step 2 using the **chmod go-w <system_command_file_location>** command.
4. Change ownership on all system command files found in step 2 using the **chown root <system_command_file_location>** command.

7.1.2.13 Verify Permissions on All Files and Directories Associated with the audit Package

Verify Permissions on All Files and Directories Associated with the audit Package

Description

This test verifies that permissions on all files and directories associated with the "audit" package. Permissions on audit binaries and configuration files that are too generous could allow an unauthorized user to gain privileges that they should not have. The permissions set by the vendor should be maintained. Any deviations from this baseline should be investigated.

Severity

0

Weight

5

Type

Content Test

Rules

Get All audit Files and Directories Have Permissions Different from RPM Database

Element

Equals "Get All audit Files and Directories Have Permissions Different from RPM Database"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Permissions more Permissive than RPM Does not exist

Remediation

To remediate failure of this policy test, restore permissions of audit package files and directories from what is expected by the RPM database.

Restoring permissions of audit package files and directories:

1. Become superuser or assume an equivalent role.
2. Run the following script:

```
Files=`/bin/rpm -V audit 2>/dev/null | /bin/grep '^.M' | /bin/
awk '{print $NF}'`; if [ ! -z "$Files" ]; then for File in $Files;
do RpmPerm=`/bin/rpm -q --queryformat "[%{FILEMODES:p
erms} %{FILENAMES}\n]" audit | /bin/egrep "[[:space:]]$Fi
le$" | /bin/awk '{print $1}'`; FilePerm=`/bin/ls -ldL "$File" | /bin/
awk '{print $1}' | /bin/sed 's/\.//'`; RpmPermRegEx=`/bin/echo
"$RpmPerm" | /bin/awk '{gsub(/[^-]/,".",$0); gsub(/^./,".",$0);
print}'`; /bin/echo "$FilePerm" | /bin/awk '$0 !~ /^'$Rpm
PermRegEx'$/{ print "'$File':\n RPM Permissions: '$Rpm
Perm'\n File Permissions: '$FilePerm'" }'; done; fi;
```

   to list files and directories have permissions different from the package given.
3. For each file and directory, run the command **chmod <permissions> <file|directory>** where **<permissions>** is equal or less permissive than RPM permissions or can use **rpm --setperms audit** command to restore file access permissions of the audit package files and directories.

7.1.2.14 Verify Permissions on All Files and Directories Associated with Packages

Verify Permissions on All Files and Directories Associated with Packages

Description

The system package management tool must verify permissions on all files and directories associated with packages. Permissions on system binaries and configuration files that are too generous could allow an unauthorized user to gain privileges that they should not have. The permissions set by the vendor should be maintained. Any deviations from this baseline should be investigated.

Severity

0

Weight

5

Type

Content Test

Rules

Get All Files and Directories Have Permissions Different from RPM Database

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Get All Files and Directories Have Permissions Different from RPM Database"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Permissions more Permissive than RPM Does not exist

Remediation

To remediate failure of this policy test, restore permission of package files and directories from what is expected by the RPM database.

Restoring permission of package files and directories:

1. Become superuser or assume an equivalent role.
2. Run the following command:

   ```
   /bin/rpm -Va 2>/dev/null | /bin/grep '^.M' | /usr/bin/awk '{print $NF}'
   ```

   to list which files on the system have permission different from what is expected by the RPM database (this could take a while).
3. For each file or directory found, run the following command to restore permission:

   ```
   Package=`/bin/rpm -qf <file_directory> 2>/dev/null` ; /bin/rpm --setperms "$Package" 2>/dev/null
   ```

7.1.2.15 Verify That wheel Is a Group of root and Other Users

Verify That wheel Is a Group of root and Other Users

Description	This test checks 'wheel' is a group of root and users in /etc/group. The su command allows a user to run a command or shell as another user. The program has been superseded by sudo, which allows for more granular control over privileged access. Normally, the su command can be executed by any user. By uncommenting the pam_wheel.so statement in /etc/pam.d/su, the su command will only allow users in the wheel group to execute su.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/group"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*wheel:[^:]+:\d+:(\S+)$/ (Flags:Multiline,Comments mode) wheel Group List Matches "^.*\broot\b.*$"
Remediation	To remediate failure of this policy test, add root user to the wheel group. **Adding root user and other users to the wheel group**: 1. Become superuser or assume an equivalent role. 2. Run the **usermod -a -G wheel *<user name>*** command to add users to the **wheel** group, where ***<user name>*** is user which are needed to run using **su** command(user **root** is required to add). For further details, please refer to: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/Security_Guide/s2-wstation-privileges-limitroot.html

7.1.2.16 Verify /etc/cron.weekly Permissions

Verify /etc/cron.weekly Permissions

Description	The /etc/cron.weekly directory contains system cron jobs that need to run on a weekly basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/cron.weekly"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^d.{3}-{6}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/cron.weekly directory. **Setting appropriate permissions and ownership on the /etc/cron.weekly directory:** 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -ld /etc/cron.weekly** command. 3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/cron.weekly** command. 4. Change ownership to **root:root** using the **chown root:root /etc/cron.weekly** command.

7.1.2.17 Verify Library File Permissions

Verify Library File Permissions

Description

This test verifies that all system library files are owned by 'root' and have permissions set to 755 or more restrictive.
Removing write permissions from system library files for everyone but the owner prevents unauthorized users from altering them.

Severity

0

Weight

5

Type

Content Test

Rules

Get Permissions of Invalid Library Files

Element

Equals "Library File Permissions"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Multiline,Case insensitive,Comments mode)
Library File Permissions Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions on library files.

Setting appropriate permissions on library files:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   /usr/bin/find -L /lib/ /lib64 /usr/lib /usr/lib64 -follow -type f -ls
   2>/dev/null | /bin/awk '($3 ~ /^.....(w|...w)/ || $5 != "root") {print
   $3,$5,$6,$NF}' 2>/dev/null
   ```

   to list invalid library files.
3. Remove write permission for group and others on invalid library files found in step 2 using the **chmod go-w <lib_file_location>** command.
4. Change ownership on invalid library files found in step 2 using the **chown root <lib_file_location>** command.

7.1.2.18 Verify That /dev/shm Partition Mounted with 'nodev'

Verify That /dev/shm Partition Mounted with 'nodev'

Description	The nodev mount option specifies that the filesystem cannot contain special devices. Since the /dev/shm filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /dev/shm.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/dev/shm[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnodev\b.*\).*$/ (Flags:Multiline,Comments mode) /dev/shm with nodev Option Exists
Remediation	To remediate failure of this policy test, set nodev option for /dev/shm partition. **Setting nodev option for /dev/shm partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/dev/shm**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/dev/shm**, then go to step 5. 5. If found, add the **nodev** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nodev /dev/shm** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2.19 Verify That .rhosts Files Do Not Exist

Verify That .rhosts Files Do Not Exist

Description	This test determines if any .rhosts files are present on the system. These files may contain unencrypted passwords which could be used to attack other systems. Examine the list of files found by this policy test very carefully and identify application dependencies and user impact before removing anything.
Severity	0
Weight	5
Type	Content Test
Rules	User Dot Files
Element	Equals "User Dot Files"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^.*/\.rhosts$/ (Flags:Multiline,Comments mode) .rhosts File Does not exist

Remediation

To remediate failure of this policy test, remove the .rhosts files in the user home directories.

Removing the .rhosts files in the user home directories:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   Users=`/bin/egrep -v "^[[:space:]]*#|^[[:space:]]*$" /etc/passwd 2>/dev/null | /bin/awk -F: '{ cmd = "/usr/bin/passwd -S " $1 " 2>/dev/null"; cmd | getline UserInfo; if ($0 !~ /^[[:space:]]*(#.*|\+.*|root|halt|sync|shutdown):/ && (UserInfo ~ /^[[:graph:]]+[[:space:]]+PS[[:space:]]+/ || (UserInfo ~ /^[[:space:]]*Unknown[[:space:]]+user\./ && $2 != "!!")) && $7 !~ /^\/sbin\/nologin$/){ print $1 ":" $6}}'`; SavedIFS="$IFS"; IFS=`/bin/echo -e "\n\b"`; for User in $Users; do UserName=`/bin/echo "$User" | /bin/awk -F: '{print $1}'`; HomeDirectory=`/bin/echo "$User" | /bin/awk -F: '{print $2}'`; /bin/ls -alL $HomeDirectory/.rhosts 2>/dev/null | awk '$1 !~ /^d/ { FileName=substr($0,index($0,"/")); print UserName, $1, $3, $4, FileName}' UserName="$UserName"; done; IFS="$SavedIFS";
   ```

   to list all **.rhosts** files.
3. Remove **.rhosts** files found in step 2 using the **rm -f <.rhosts_file_name>** command.

For further details, please run the command **man rm** to read man page.

7.1.2.20 Verify /etc/grub.conf Permissions

Verify /etc/grub.conf Permissions

Description

This test verifies that the 'root' user owns /etc/grub.conf and permissions are equal to 700 or more restrictive.
To help protect the GRUB configuration from unauthorized changes, only the 'root' user should have read and write access to the grub.conf file.

Severity

0

Weight

5

Type

Content Test

Rules

/etc/grub.conf Permissions

Element

Equals "/etc/grub.conf Permissions"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^(\S+).?[\ \t]+(\S+)[\ \t]+(\S+)[\ \t]+/etc/grub\.conf[\ \t]*$/ (Flags:Multiline,Comments mode)
Permission Matches "^-...-{6}.*" AND
User Equals "root" AND
Group Equals "root"

Remediation

To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/grub.conf file.

Setting appropriate permissions and ownership on the /etc/grub.conf file:

1. Become superuser or assume an equivalent role.
2. Check the permissions and ownership of the file using the **ls -lL /etc/grub.conf** command.
3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/grub.conf** command.
4. Change ownership to **root** using the **chown root:root /etc/grub.conf** command.

7.1.2.21 Verify /etc/cron.hourly Permissions

Verify /etc/cron.hourly Permissions

Description	This directory contains system cron jobs that need to run on an hourly basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/cron.hourly"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^d.{3}-{6}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/cron.hourly directory. **Setting appropriate permissions and ownership on the /etc/cron.hourly directory:** 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -ld /etc/cron.hourly** command. 3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/cron.hourly** command. 4. Change ownership to **root:root** using the **chown root:root /etc/cron.hourly** command.

7.1.2.22 Verify /etc/cron.daily Permissions

Verify /etc/cron.daily Permissions

Description	The /etc/cron.daily directory contains system cron jobs that need to run on a daily basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/cron.daily"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^d.{3}-{6}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/cron.daily directory. **Setting appropriate permissions and ownership on the /etc/cron.daily directory:** 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -ld /etc/cron.daily** command. 3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/cron.daily** command. 4. Change ownership to **root:root** using the **chown root:root /etc/cron.daily** command.

7.1.2.23 Verify /etc/cron.monthly Permissions

Verify /etc/cron.monthly Permissions

Description	The /etc/cron.monthly directory contains system cron jobs that need to run on a weekly basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/cron.monthly"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^d.{3}-{6}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/cron.monthly directory. **Setting appropriate permissions and ownership on the /etc/cron.monthly directory:** 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -ld /etc/cron.monthly** command. 3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/cron.monthly** command. 4. Change ownership to **root:root** using the **chown root:root /etc/cron.monthly** command.

7.1.2.24 Verify /etc/cron.d Permissions

Verify /etc/cron.d Permissions

Description	The /etc/cron.d directory contains system cron jobs that need to run in a similar manner to the hourly, daily weekly and monthly jobs from /etc/crontab, but require more granular control as to when they run. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/cron.d"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^d.{3}-{6}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/cron.d directory. **Setting appropriate permissions and ownership on the /etc/cron.d directory**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the directory using the **ls -ldL /etc/cron.d** command. 3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/cron.d** command. 4. Change ownership to **root:root** using the **chown root:root /etc/cron.d** command.

7.1.2.25 Verify That the ntp Daemon Is Running as an Unprivileged User

Verify That the ntp Daemon Is Running as an Unprivileged User

<table>
<tr><td><b>Description</b></td><td>The Network Time Protocol (NTP) is designed to synchronize system clocks across a variety of systems and use a source that is highly accurate. It is recommended that physical systems and virtual guests lacking direct access to the physical host's clock be configured as NTP clients to synchronize their clocks (especially to support time sensitive security mechanisms like Kerberos). This also ensures log files have consistent time records across the enterprise, which aids in forensic investigations.</td></tr>
<tr><td><b>Severity</b></td><td>0</td></tr>
<tr><td><b>Weight</b></td><td>5</td></tr>
<tr><td><b>Type</b></td><td>Content Test</td></tr>
<tr><td><b>Rules</b></td><td>System Configuration Files</td></tr>
<tr><td><b>Excluded Nodes</b></td><td>Red Hat Enterprise Linux Server 7
Red Hat Enterprise Linux Server 6
Red Hat Enterprise Linux Server 5</td></tr>
<tr><td><b>Element</b></td><td>Equals "/etc/sysconfig/ntpd"</td></tr>
<tr><td><b>Version conditions</b></td><td>If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*OPTIONS[\ \t]*=[\ \t]*".*-u[\ \t]+(\w+:\w+)(?:"|[\ \t]+[^"\#]*")[\ \t]*(?:$|\#)
/ (Flags:Multiline,Comments mode)
ntp Daemon Equals "ntp:ntp"</td></tr>
<tr><td><b>Remediation</b></td><td>To remediate the failure of this policy test, set user parameters to ensure that NTP daemon is running as an unprivileged user.
<b>Setting user parameters to ensure that NTP daemon is running as an unprivileged user</b>:
1. Become a superuser or assume an equivalent role.
2. If ntp account and ntp group dedicated to unprivileged user doesn't exist, add them to system:
• Run the following command to add new group: <b>groupadd &lt;</b>group_name<b>&gt; -g &lt;</b>value<b>&gt;</b>
• Run the following command to add new account: <b>useradd &lt;</b>account_name<b>&gt; -s /usr/sbin/nologin -u &lt;</b>value<b>&gt; -g &lt;</b>value<b>&gt;</b>
<i>Note: The <value> in the above commands is userid and groupid, you can choose any number which is less than 500 and not duplicated with another userid - groupid.</i>
3. Open <b>/etc/sysconfig/ntpd</b> file.
4. Find the line that contains <b>OPTIONS</b> entry.
5. Uncomment or change it to <b>OPTIONS="-u ntp:ntp -p /var/run/ntpd.pid"</b> or add if not found.
6. Save and close the file.
For further details, please run the command <b>man ntpd</b> to read man page.</td></tr>
</table>

7.1.2.26 Verify That /tmp Partition Mounted with 'nodev'

Verify That /tmp Partition Mounted with 'nodev'

Description	The nodev mount option specifies that the filesystem cannot contain special devices. Since the /tmp filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /tmp.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/tmp[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnodev\b.*\).*$/ (Flags:Multiline,Comments mode) /tmp with nodev Option Exists
Remediation	To remediate failure of this policy test, set nodev option for /tmp partition. **Setting nodev option for /tmp partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/tmp**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/tmp**, then go to step 5. 5. If found, add the **nodev** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nodev /tmp** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2.27 Verify That at Least One of AllowUsers, AllowGroups, DenyUsers, DenyGroups Option Is Leveraged

Verify That at Least One of AllowUsers, AllowGroups, DenyUsers, DenyGroups Option Is Leveraged

Description

There are several options available to limit which users and group can access the system via SSH. It is recommended that at least of the following options be leveraged: AllowUsers, AllowGroups, DenyUsers, DenyGroups.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/ssh/sshd_config"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*(?:AllowUsers|AllowGroups|DenyUsers|DenyGroups)[\ \t]+\w+.*$/ (Flags:Multiline,Case insensitive,Comments mode)
Access via SSH Setting Exists

Remediation

To remediate failure of this policy test, configure the SSH server to limit which users and group can access the system via SSH.

Configuring the SSH server to limit which users and group can access the system via SSH:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Adding at least of the following options:

   AllowUsers <user_list>
   AllowGroups <group_list>
   DenyUsers <user_list>
   DenyGroups <group_list>

   where **<user_list>** and **<group_list>** is a list of user name or group name patterns, separated by comma.
4. Save the file.
5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

For further details, please run the command **man sshd_config** to read man page.

7.1.2.28 Verify That the fs.suid_dumpable Parameter Is Set to 0

Verify That the fs.suid_dumpable Parameter Is Set to 0

Description

This test verify That fs.suid_dumpable is set to 0. When suid_dumpable is set to 0, a core dump will not be produced for a process which has changed credentials (by calling seteuid(2), setgid(2), or similar, or by executing a set-user-ID or set-group-ID program) or whose binary does not have read permission enabled.

Severity

0

Weight

5

Type

Content Test

Rules

Kernel Parameters

Element

Equals "Kernel Parameters"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*/proc/sys/fs/suid_dumpable[\ \t]*:[\ \t]*(\d+)[\ \t]*$/ (Flags:Multiline,Comments mode)
fs.suid_dumpable Equals 0

Remediation

To remediate failure of this policy test, set fs.suid.dumpable to disable core dumps in order to prevent suid programs from dumping core.

Setting fs.suid_dumpable to disable core dumps:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysctl.conf** file.
3. Find the lines **fs.suid_dumpable = <value>**.
4. Set the **<value>** to **0** and save the file.
5. If there no line setting **fs.suid_dumplable**, add the following line:

   fs.suid_dumpable = 0

   at the end of the file and save the file.
6. Run the **sysctl -p** command to apply the change.

For further details, please run the command **man sysctl.conf** to read man page.

7.1.2.29 Verify That /home Partition Mounted with 'nodev'

Verify That /home Partition Mounted with 'nodev'

Description	The nodev mount option specifies that the filesystem cannot contain special devices. Since the /home filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /home.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/home[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnodev\b.*\).*$/ (Flags:Multiline,Comments mode) /home with nodev Option Exists
Remediation	To remediate failure of this policy test, set nodev option for /home partition. **Setting nodev option for /home partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/home**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/home**, then go to step 5. 5. If found, add the **nodev** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nodev /home** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2.30 Verify That /dev/shm Partition Mounted with 'nosuid'

Verify That /dev/shm Partition Mounted with 'nosuid'

Description	The nosuid mount option specifies that the /dev/shm (temporary filesystem stored in memory) will not execute setuid and setgid on executable programs as such, but rather execute them with the uid and gid of the user executing the program. Setting this option on a file system prevents users from introducing privileged programs onto the system and allowing non-root users to execute them.
Severity	0
Weight	5
Type	Content Test
Rules	File Systems Mounted
Element	Equals "File Systems Mounted"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/dev/shm[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnosuid\b.*\).*$/ (Flags:Multiline,Comments mode) /dev/shm with nosuid Option Exists
Remediation	To remediate failure of this policy test, set nosuid option for /dev/shm partition. **Setting nosuid option for /dev/shm partition**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/fstab** file. 3. Find the line with options for **/dev/shm**. 4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/dev/shm**, then go to step 5. 5. If found, add the **nosuid** option to the fourth field, using a comma to separate from other options. 6. Remount partition by using the **mount -o remount,nosuid /dev/shm** command. For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems: http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2.31 Verify That /dev/shm Partition Mounted with 'noexec'

Verify That /dev/shm Partition Mounted with 'noexec'

Description

Set noexec on the shared memory partition to prevent programs from executing from there. Setting this option on a file system prevents users from executing programs from shared memory. This deters users from introducing potentially malicious software on the system.

Severity

0

Weight

5

Type

Content Test

Rules

File Systems Mounted

Element

Equals "File Systems Mounted"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*[^\#&&\S]+[\ \t]+on[\ \t]+/dev/shm[\ \t]+type[\ \t]+[^\#&&\S]+[\ \t]+\([^\#&&\S]*\bnoexec\b.*\).*$/ (Flags:Multiline,Comments mode)
/dev/shm with noexec Option Exists

Remediation

To remediate failure of this policy test, set noexec option for /dev/shm partition.

Setting noexec option for /dev/shm partition:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/fstab** file.
3. Find the line with options for **/dev/shm**.
4. If not found, use the Logical Volume Manager (LVM) to create a separate partition for **/dev/shm**, then go to step 5.
5. If found, add the **noexec** option to the fourth field, using a comma to separate from other options.
6. Remount partition by using the **mount -o remount,noexec /dev/shm** command.

For further details, see the guidance on the Logical Volume Manager (LVM) for more information on repartitioning filesystems:

http://tldp.org/HOWTO/LVM-HOWTO/

7.1.2.32 Verify That the PROMPT Parameter in /etc/sysconfig/init Is Set to no

Verify That the PROMPT Parameter in /etc/sysconfig/init Is Set to no

Description

The PROMPT option provides console users the ability to interactively boot the system and select which services to start on boot. Turn off the PROMPT option on the console to prevent console users from potentially overriding established security settings.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/sysconfig/init"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*PROMPT=(\w+)[\ \t]*(?:\#|$)/ (Flags:Multiline,Comments mode)
PROMPT parameter Equals "no"

Remediation

To remediate failure of this policy test, turn off the PROMPT option.

Turning off the PROMPT option:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysconfig/init** file.
3. Find the line **PROMPT=<parameter>**.
4. If found, then set **<parameter>** to **no** and save the file.
5. If not found, then add the **PROMPT=no** line to the file and save it.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/ch-sysconfig.html#s1-sysconfig-files.

7.1.2.33 Verify That PermitUserEnvironment Option Is Set to no

Verify That PermitUserEnvironment Option Is Set to no

Description	The PermitUserEnvironment option allows users to present environment options to the ssh daemon. Permitting users the ability to set environment variables through the SSH daemon could potentially allow users to bypass security controls (e.g. setting an execution path that has ssh executing trojan programs)
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*PermitUserEnvironment[\ \t]+(\S+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) PermitUserEnvironment Not equal "yes"
Remediation	To remediate failure of this policy test, configure the SSH server to disable environment processing. **Configuring the SSH server to disable environment processing:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **PermitUserEnvironment <value>**. 4. Set **<value>** to **no** and save the file. 5. Run the **service sshd restart** command to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.

7.1.2.34 Verify /etc/shadow Permissions

Verify /etc/shadow Permissions

Description

This test verifies that the 'root' user owns /etc/shadow and permissions are equal to 000. It is worthwhile to periodically check these file permissions as there have been package defects that changed /etc/shadow permissions to 000.

Severity

0

Weight

5

Type

Attribute Test

Rules

System Configuration Files

Element

Equals "/etc/shadow"

Version conditions

Action if missing:Pass
User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND
Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND
Permissions Matches "^-{10}.*$"

Remediation

To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/shadow file.

Setting appropriate permissions and ownership on the /etc/shadow file:

1. Become superuser or assume an equivalent role.
2. Check the permissions and ownership of the file using the **ls -lL /etc/shadow** command.
3. Change permissions to **000** using the **chmod 000 /etc/shadow** command.
4. Change ownership to **root:root** using the **chown root:root /etc/shadow** command.

7.1.2.35 Verify User .netrc Files Permissions

Verify User .netrc Files Permissions

Description

.netrc files may contain unencrypted passwords which may be used to attack other systems. This test verifies that the permissions of .netrc files are equal to 700 or more restrictive.

Severity

0

Weight

5

Type

Content Test

Rules

User Dot Files

Element

Equals "User Dot Files"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^\S+[\ \t]+(?!.{4}-{6})\S+[\ \t]+.*/\.netrc$/ (Flags:Multiline,Comments mode)
.netrc Permissions Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions on .netrc files.

Setting appropriate permissions on .netrc files:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   Users=`/bin/egrep -v "^[[:space:]]*#|^[[:space:]]*$" /etc/pass
   wd 2>/dev/null | /bin/awk -F: '{ cmd = "/usr/bin/passwd -S "
   $1 " 2>/dev/null"; cmd | getline UserInfo; if ($0 !~ /^[[:space:
   ]]*(#.*|\+.*|root|halt|sync|shutdown):/ && (UserInfo ~ /^[[:g
   raph:]]+[[:space:]]+PS[[:space:]]+/ || (UserInfo ~ /^[[:space:
   ]]*Unknown[[:space:]]+user\./ && $2 != "!!")) && $7 !~ /^\/sbin
   \/nologin$/){ print $1 ":" $6}}'`; SavedIFS="$IFS"; IFS=`/bin/
   echo -e "\n\b"`; for User in $Users; do UserName=`/bin/echo
   "$User" | /bin/awk -F: '{print $1}'`; HomeDirectory=`/bin/echo
   "$User" | /bin/awk -F: '{print $2}'`; /bin/ls -alL $HomeDirec
   tory/.netrc 2>/dev/null | awk '($1 !~ /^d/ && $1 !~ /....------/)
   { FileName=substr($0,index($0,"/")); print UserName, $1, $3,
   $4, FileName}' UserName="$UserName"; done; IFS="$Save
   dIFS";
   ```

   to list files which have inappropriate permissions.
3. Set permissions on **.netrc** files found in step 2 to 700 or more restrictive using the **chmod go-rwx <.netrc_file_name>** command.

For further details, please refer to:

http://www.redhat.com/mirrors/LDP/LDP/GNU-Linux-Tools-Summary/html/file-permissions.html

7.1.2.36 Verify Home Directories Ownership

Verify Home Directories Ownership

Description	This test checks that all home directories are owned by the user associated with them. In conjunction with proper permissions, correct ownership prevents unauthorized change.
Severity	0
Weight	5
Type	Content Test
Rules	User Home Directories
Element	Equals "User Home Directories"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^UserName=(?!nfsnobody[\ \t])\S+[\ \t]+UserID=([5-9]\d{2}|0*\d{4,})[\ \t]+.*Owner=(?!\1[\ \t])\S+[\ \t]+HomeDirExisted=yes$/ (Flags:Multiline,Comments mode) Home Directories Ownership Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate ownership on the home directory of each account.

Setting appropriate ownership on the home directory of each account:

1. Become superuser or assume an equivalent role.
2. Run the script:

```
Users=`/bin/cat /etc/passwd 2>/dev/null | /bin/egrep -v "^[[:
space:]]*(#.*|\+.*|nfsnobody):" | /bin/awk -F: '$3 >= 500 {prin
t}'`; IFS=`/bin/echo -en "\n\b"`; SavedIFS="$IFS"; IFS=`/bin/
echo -en "\n\b"`; for User in $Users; do UserAcct=`/bin/echo
$User | /bin/cut -d":" -f1`; UserHome=`/bin/echo $User | /bin/
cut -d":" -f6`; if [ -d "$UserHome" ] && [ "$UserHome" != "/" ];
then Owner=`/usr/bin/stat -c %U $UserHome 2>/dev/null`;if
[ "$Owner" != "$UserAcct" ]; then /bin/echo -e "The [ $User
Acct ] user has [ $UserHome ] home directory with invalid
ownership of [ $Owner ]";fi;fi;done;IFS="$SavedIFS"
```

   to list users of which home directory is not owned by the assigned user except the "**/**" directory.
3. For each user found in step 2, run the **chown <assigned_user> <home_dir_location>** command to set owner of the home directory to the assigned user.

Note: If the script output returns a local account that duplicate name with others, recommend that you should remove or comment it out.

7.1.2.37 Verify /etc/crontab Permissions

Verify /etc/crontab Permissions

Description

The /etc/crontab file is used by cron to control its own jobs. The commands in this item make here sure that root is the user and group owner of the file and is the only user that can read and write the file.

Severity

0

Weight

5

Type

Attribute Test

Rules

System Configuration Files

Element

Equals "/etc/crontab"

Version conditions

Action if missing:Pass
User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND
Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND
Permissions Matches "^-.{3}-{6}.*$"

Remediation

To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/crontab file.

Setting appropriate permissions and ownership on the /etc/crontab file:

1. Become superuser or assume an equivalent role.
2. Check the permissions and ownership of the file using the **ls -lL /etc/crontab** command.
3. Change permissions to **700** or more restrictive using the **chmod go-rwx /etc/crontab** command.
4. Change ownership to **root:root** using the **chown root:root /etc/crontab** command.

7.1.2.38 Verify /etc/group Permissions

Verify /etc/group Permissions

Description	This test verifies that the 'root' user owns the /etc/group file and permissions are equal to 644 or more restrictive. Setting the recommended permissions allows users to view the file, but only 'root' has write access.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/group"
Version conditions	Action if missing:Pass User Matches "^root\s\(\d+\)\s*$" AND Group Matches "^root\s\(\d+\)\s*$" AND Permissions Matches "^-..-.-{2}.-{2}.*"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/group file. **Setting appropriate permissions and ownership on the /etc/group file**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file by using the **ls -lL /etc/group** command. 3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/group** command. 4. Change ownership to **root:root** using the **chown root:root /etc/group** command.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-wx"
PermsRegex="-..-.--.--"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/group"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0000881
# AR_TEST_NAME = Verify /etc/group Permissions
```

Post Remediation Category *None*

Remediated Elements *None*

Post Remediation Steps No additional Post Remediation steps

7.1.2.39 User Home Directories Should Be Mode 750 or More Restrictive

User Home Directories Should Be Mode 750 or More Restrictive

Description

This test verifies that user home directories are not group-writable and only members of the same group have read and execute access.
This reduces the risk posed by malicious users and allows for users to define access control at their discretion. Carefully consider the impact that any configuration changes to home directory permissions will have in your environment.

Severity

0

Weight

5

Type

Content Test

Rules

User Home Directories Permissions

Element

Equals "User Home Directories Permissions"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*d(?!.{4}-.-{3})\S+[\ \t]+.*$/ (Flags:Multiline,Comments mode)
Home Directory Permissions Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions on user home directories.

Setting appropriate permissions on user home directories:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   Users=`/bin/cat /etc/shadow 2>/dev/null | /bin/sort -u | /bin/
   egrep -v "^[[:space:]]*(#.*|\+.*|root|halt|sync|shutdown):"
   | /bin/awk -F: '$2 !~ /^(\*|\!|\!\!|\!\*)$/ {print $1}'`; for User in
   $Users; do UserHome=`/bin/awk -F: '$1 ~ /^'$User'$/ && $7 !
   = "/sbin/nologin" {print $6}' /etc/passwd`; if [ "$UserHome" !=
   "/" ]; then /bin/ls -ldL "$UserHome" 2>/dev/null | /bin/awk '$1 !
   ~ /d....-.---/ {print $1,$NF}'; fi; done;
   ```

   to list user home directories which have inappropriate permissions.
3. Set permissions on user home directories found in step 2 to **750** or more restrictive using the **chmod g-w,o-rwx <user_home_directory>** command.

For further details, please refer to:

http://www.redhat.com/mirrors/LDP/LDP/GNU-Linux-Tools-Summary/html/file-permissions.html

7.1.2.40 No User Dot-files Should Be World-writable

No User Dot-files Are Group/World-writable

Description

This test verifies that user dot-files are not group/world-writable. Group/world-writable user configuration files may enable malicious users to steal or modify other users' data or to gain another user's system privileges. The system administrator should examine any files found by this policy test.

Severity

0

Weight

5

Type

Content Test

Rules

User Dot Files

Element

Equals "User Dot Files"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^\S+[\ \t]+(?:.{5}|.{8})w.*$/ (Flags:Multiline,Comments mode)
Dot-file Permissions Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions on user dot-files.

Setting appropriate permissions on user dot-files:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   Users=`/bin/egrep -v "^[[:space:]]*#|^[[:space:]]*$" /etc/pass
   wd 2>/dev/null | /bin/awk -F: '{ cmd = "/usr/bin/passwd -S "
   $1 " 2>/dev/null"; cmd | getline UserInfo; if ($0 !~ /^[[:space:
   ]]*(#.*|\+.*|root|halt|sync|shutdown):/ && (UserInfo ~ /^[[:g
   raph:]]+[[:space:]]+PS[[:space:]]+/ || (UserInfo ~ /^[[:space:
   ]]*Unknown[[:space:]]+user\./ && $2 != "!!")) && $7 !~ /^\/sbin
   \/nologin$/){ print $1 ":" $6}}'`; SavedIFS="$IFS";  IFS=`/bin/
   echo -e "\n\b"`; for User in $Users; do UserName=`/bin/echo
   "$User" | /bin/awk -F: '{print $1}'`; HomeDirectory=`/bin/echo
   "$User" | /bin/awk -F: '{print $2}'`; /bin/ls -alLd $HomeDirec
   tory/.[A-Za-z0-9]* 2>/dev/null | /bin/awk '$1 !~ /^d/ && $1 ~ /
   (.....|........)w/ { FileName=substr($0,index($0,"/")); print User
   Name, $1, $3, $4, FileName}' UserName="$UserName"; done;
   IFS="$SavedIFS"
   ```

   to list user dot-files which have inappropriate permissions.
3. Remove group world-writable on the user dot-files found in step 2 using the **chmod go-w <user_dot_file>** command.

7.1.2.41 Verify No Group/World-writable Directory in $PATH

Verify No Group/World-writable Directory in $PATH

Description

This test verifies that there are no groups or world-writable directories included in root's executable path.
Remediating failure of this test can help reduce the risk of an attacker gaining superuser access by forcing a root user to execute a malicious executable.

Severity

0

Weight

5

Type

Content Test

Rules

Verify No Group/World-writable Directory in $PATH

Element

Equals "Group/World-writable Directory"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.*/ (Flags:Case insensitive)
Group/World-writable Directories Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions on the group/world writable directories in the PATH.

Setting appropriate permissions on the group/world writable directories in the PATH:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   /usr/bin/find `/bin/echo $PATH|/bin/sed "s/:/ /g"` -prune -type
   d -ls 2>/dev/null | /bin/awk '$3 ~ /^(.....w|........w)/ || $5 != "root
   " {OFS=":";print $3,$5,$6,substr($0,index($0,"/"),length($0))}'
   ```

   to list directories which have inappropriate permissions on the group/world writable directories in the PATH.
3. Set permissions on directories in the PATH found in step 2 to remove writable permissions of group and others using the **chmod go-w <directory_location>** command.
4. Set the owner on directories in the PATH found in step 2 to **root** using the **chown root <directory_location>** command.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Description="Group/World-writable Directories in $PATH"
Perms="go-w"
PermsRegex="d....-..-."
Owner=""
OwnersRegex=""
Group=""
GroupsRegex=""
ExistingElement="Pass"
FileEntries="$(/usr/bin/find `/bin/echo $PATH | /bin/sed "s/:/ /
g"` -prune \
    -type d \( -perm -g+w -o -perm -o+w \) -ls 2>/dev/null | \
    /bin/awk '{OFS=":";print $3,$5,
$6,substr($0,index($0,"/"),length($0))}')"

SavedIFS=$IFS
IFS=$(/bin/echo -e "\n\b")
for FileEntry in $FileEntries; do
    unset Permissions
    unset PermsLog
    unset OwnerLog
    unset GroupLog
    FileName=`/bin/echo "$FileEntry" | /bin/cut -d: -f4-`
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk -F: '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk -F: '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk -F: '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        FailureFiles="$FailureFiles\n[$Permissions] to
 [$FileName]"
        FailureFiles="$FailureFiles file/directory"
    else
        SuccessFiles="$SuccessFiles\n[$Permissions] to
 [$FileName]"
        SuccessFiles="$SuccessFiles file/directory"
    fi
done
IFS=$SavedIFS
if [ -n "$FailureFiles" ]; then
    /bin/echo -e "FAILURE-7001: Could not apply permissions:
$FailureFiles"
    if [ -n "$SuccessFiles" ]; then
        /bin/echo -e "\nApplied permissions:$SuccessFiles"
    fi
    exit 7001
else
    if [ -n "$SuccessFiles" ]; then
        /bin/echo -e "SUCCESS-7001: Applied permissions:
$SuccessFiles"
    else
        if [ "$ExistingElement" == "Pass" ]; then
            /bin/echo -e "SUCCESS-7001: [$Description] file/
directory with"\
                "wrong permissions does not exist"
        else
            /bin/echo -e "FAILURE-7001: [$Description] file/
directory with"\
                "wrong permissions does not exist"
            exit 7001
        fi
    fi
fi
exit 0

# AR_ACTION = RHEL_MULTIPLE_FILES_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0000916
# AR_TEST_NAME = Verify No Group/World-writable Directory in
 $PATH
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.42 Limit Access to the Root Account from su

Limit Access to the Root Account from su

<table>
<tr><td><b>Description</b></td><td>This test checks /etc/pam.d/su to verify that only members of the wheel group have privileges enabling them to become 'root' by using the 'su' command and entering the 'root' password.
It is security best practice to carefully restrict access to administrator accounts.</td></tr>
<tr><td><b>Severity</b></td><td>0</td></tr>
<tr><td><b>Weight</b></td><td>5</td></tr>
<tr><td><b>Type</b></td><td>Content Test</td></tr>
<tr><td><b>Rules</b></td><td>System Configuration Files</td></tr>
<tr><td><b>Element</b></td><td>Equals "/etc/pam.d/su"</td></tr>
<tr><td><b>Version conditions</b></td><td>If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+required[\ \t]+[^\#]*pam_wheel\.so[\ \t]+use_uid[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode)
Access to su Limited to Wheel Members Exists</td></tr>
<tr><td><b>Remediation</b></td><td>To remediate failure of this policy test, configure pam.d to limit access to the 'root' account from super user to users within the wheel group.
<b>Configuring pam.d to limit access to the 'root' account from super user</b>:
1. Become superuser or assume an equivalent role.
2. Open the <b>/etc/pam.d/su</b> file.
3. Add the line that contains <b>auth required pam_wheel.so use_uid</b> to the file and save it.
<b>Note</b>: You must first have a user configured in the wheel group before making the change or else it will not be possible to su to root.
For further details, please refer to:
<b>RHEL 5</b>:
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/ch-sec-network.html#s1-wstation-privileges
<b>RHEL 6</b>:
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html-single/Deployment_Guide
<b>RHEL 7</b>:
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System_Administrators_Guide</td></tr>
<tr><td><b>Command Line</b></td><td><code>/bin/sh $(ScriptFile.sh)</code></td></tr>
</table>

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/pam.d/su"
Line="auth required pam_wheel.so use_uid"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0000926
# AR_TEST_NAME = Limit Access to the Root Account from su
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.43 Verify That umask Daemon Is at Least 027

Verify That umask Daemon Is at Least 027

Description

This test verifies that the default umask setting for the system is at least 027.
It is important to configure the system default umask in a stringent manner in order to prevent daemon processes (such as the syslog daemon) from creating world-writable files by default.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/sysconfig/init"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*umask[\ \t](?![\ \t]*0*[0-7]?[2367]7[\ \t]*(?:$|\#)).*/ (Flags:Multiline,Comments mode)
umask Setting Deviation Does not exist

Remediation

To remediate failure of this policy test, configure the functions file to set daemon umask to at least 027.

Configuring the functions file to set daemon umask to at least 027:

1. Become superuser or assume an equivalent role.
2. Open **/etc/sysconfig/init** file.
3. Find the line **umask <value>**.
4. If found, replace the **<value>** to **xy7**, where **0=< x =< 7**; **y=2,3,6,7**.
5. If not found, add the line **umask xy7** to the file with **x**, **y** as the above.
6. Save the file.

For further details, please run the command **man umask** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/init.d/functions"
ParameterName="umask"
SeparateSymbol=" "
Value="027"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0000946
# AR_TEST_NAME = Verify That umask Daemon Is at Least 027
```

Post Remediation Category	*None*
Remediated Elements	/etc/init.d/functions
Post Remediation Steps	No additional Post Remediation steps

7.1.2.44 Verify That sshd_config Disables PermitRootLogin

Verify That sshd_config Disables PermitRootLogin

Description

This test verifies that PermitRootLogin option is disabled.
Users should access the system using a non-privileged user in conjunction with an authorized mechanism, such as su or sudo, in order to gain root access. This provides for better audit trail capabilities.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/ssh/sshd_config"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*PermitRootLogin[\ \t]+(\w+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode)
SSH Server PermitRootLogin Setting Equals "no"

Remediation

To remediate failure of this policy test, configure the SSH server to disable root login via SSH.

Configuring the SSH server to disable root login via SSH:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Find the line **PermitRootLogin <value>**.
4. Set **<value>** to **no** and save the file.
5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

For further details, please run the command **man sshd_config** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
ParameterName="PermitRootLogin"
SeparateSymbol=" "
Value="no"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
                "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '{IGNORECASE=1;} $1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}'
 ${FileName} \
    2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '{IGNORECASE=1;} ($1 ~ /
^[[:space:]]*'"$ParameterName"'[[:space:]]*$/) \
        {$0 = Line;}{print}'
 Line="${ParameterName}${SeparateSymbol}${Value}" \
        ${BackupName} > ${FileName}) 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]"\
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName]"\
        "parameter changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> $FileName) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line
 to"\
            "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]"\
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_CASE_INSENSITIVE
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0003251
# AR_TEST_NAME = Verify That sshd_config Disables PermitRootLogin


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>pkill -HUP sshd</b> or <b>/
sbin/service sshd restart</b> commands to restart the <b>sshd </
b>service.</li></ol>
```

Post Remediation Category `Other`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

7.1.2.45 Verify /etc/at.allow Permissions

Verify /etc/at.allow Permissions

Description	The at daemon works with the cron daemon to allow non-privileged users to submit one time only jobs at their convenience. There are two files that control at: /etc/at.allow and /etc/at.deny. If /etc/at.allow exists, then users listed in the file are the only ones that can create at jobs. If /etc/at.allow does not exist and /etc/at.deny does exist, then any user on the system, with the exception of those listed in /etc/at.deny, are allowed to execute at jobs. An empty /etc/at.deny file allows any user to create at jobs. If neither /etc/at.allow nor /etc/at.deny exist, then only superuser can create at jobs. The commands below remove the /etc/at.deny file and create an empty /etc/at.allow file that can only be read and modified by user and group root.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/at.allow"
Version conditions	Action if missing:Fail User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^-.{2}-{7}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/at.allow file. **Setting appropriate permissions and ownership on the /etc/at.allow file:** 1. Become superuser or assume an equivalent role. 2. Run the **touch /etc/at.allow** command to create the **/etc/at.allow** file if it does not exist. 3. Check the permissions and ownership of the file using the **ls -lL /etc/at.allow** command. 4. Change permissions to **600** or more restrictive using the **chmod u-x,go-rwx /etc/at.allow** command. 5. Change ownership to **root:root** using the **chown root:root /etc/at.allow** command.

7.1.2.46 Verify /etc/cron.allow Permissions

Verify /etc/cron.allow Permissions

Description	This test verifies that the /etc/cron.allow file has owned and group owned by root, and permissions of 600 or more restrictive. This gives root read and write permissions while all other users have no access.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/cron.allow"
Version conditions	Action if missing:Fail User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^-.{2}-{7}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/cron.allow file. **Setting appropriate permissions and ownership on the /etc/cron.allow file:** 1. Become superuser or assume an equivalent role. 2. Run the **touch /etc/cron.allow** command to create the **/etc/cron.allow** file if it does not exist. 3. Check the permissions and ownership of the file using the **ls -lL /etc/cron.allow** command. 4. Change permissions to **600** or more restrictive using the **chmod u-x,go-rwx /etc/cron.allow** command. 5. Change ownership to **root:root** using the **chown root:root /etc/cron.allow** command.

7.1.2.47 Verify /etc/motd Permissions

Verify /etc/motd Permissions

Description	This test verifies that the 'root' user owns /etc/motd and permissions are equal to 644 or more restrictive. After configuring a login banner for console access, it is important to protect the file from unauthorized changes by granting only the 'root' user write access.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/motd"
Version conditions	Action if missing:Pass User Matches "^root\s\(\d+\)\s*$" AND Group Matches "^root\s\(\d+\)\s*$" AND Permissions Matches "^-..-.-{2}.-{2}.*"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/motd file. **Setting appropriate permissions and ownership of the /etc/motd file**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -lL /etc/motd** command. 3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/motd** command. 4. Change ownership to **root:root** using the **chown root:root /etc/motd** command.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-wx"
PermsRegex="-..-.--.--"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/motd"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003390
# AR_TEST_NAME = Verify /etc/motd Permissions
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.48 Verify /etc/issue Permissions

Verify /etc/issue Permissions

Description

This test verifies that the 'root' user owns /etc/issue and permissions are equal to 644 or more restrictive.
After configuring a login banner for console access, it is important to protect the file from unauthorized changes by granting only the 'root' user write access.

Severity

0

Weight

5

Type

Attribute Test

Rules

System Configuration Files

Element

Equals "/etc/issue"

Version conditions

Action if missing:Pass
User Matches "^root\s\(\d+\)\s*$" AND
Group Matches "^root\s\(\d+\)\s*$" AND
Permissions Matches "^-..-.-{2}.-{2}.*"

Remediation

To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/issue file.

Setting appropriate permissions and ownership of the /etc/issue file:

1. Become superuser or assume an equivalent role.
2. Check the permissions and ownership of the file using the **ls -lL /etc/issue** command.
3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/issue** command.
4. Change ownership to **root:root** using the **chown root:root /etc/issue** command.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-wx"
PermsRegex="-..-.--.--"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/issue"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003392
# AR_TEST_NAME = Verify /etc/issue Permissions
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.49 Verify /etc/issue.net Permissions

Verify /etc/issue.net Permissions

Description

This test verifies that the 'root' user owns /etc/issue.net and permissions are equal to 644 or more restrictive.
After configuring a login banner for network access, it is important to protect the file from unauthorized changes by granting only the 'root' user write access.

Severity

0

Weight

5

Type

Attribute Test

Rules

System Configuration Files

Element

Equals "/etc/issue.net"

Version conditions

Action if missing:Pass
User Matches "^root\s\(\d+\)\s*$" AND
Group Matches "^root\s\(\d+\)\s*$" AND
Permissions Matches "^-..-.-{2}.-{2}.*"

Remediation

To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/issue.net file.

Setting appropriate permissions and ownership on the /etc/issue.net file:

1. Become superuser or assume an equivalent role.
2. Check the permissions and ownership of the file using the **ls -lL /etc/issue.net** command.
3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/issue.net** command.
4. Change ownership to **root:root** using the **chown root:root /etc/issue.net** com mand.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-wx"
PermsRegex="-..-.--.--"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/issue.net"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003394
# AR_TEST_NAME = Verify /etc/issue.net Permissions
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.50 Verify No '.' in $PATH

Verify No '.' in $PATH

Description

This test verifies that the root's PATH variable does not contain a '.' or '::' or starts or ends with ':'. That prevents an attacker from gaining superuser access by forcing an administrator operating as root to execute a Trojan horse program.

Severity

0

Weight

5

Type

Content Test

Rules

Verify No '.' in $PATH

Element

Equals "$PATH variable"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /(?:^|:)\.?(?::\.?|$)/ (Flags:Multiline,Case insensitive,Comments mode)
'.' in $PATH Does not exist

Remediation

To remediate failure of this policy test, exclude ":" or ".:" at the first part and the last part, and "::" or ":.:" at the mid part of the root PATH variable.

Excluding ":" or ".:" at the first part and the last part, and "::" or ":.:" at the mid part of the root PATH variable:

1. Become superuser or assume an equivalent role.
2. Run the **egrep "^[[:space:]]*root:" /etc/passwd | awk -F: '{print $6}'** command to list root home directory.
3. Check the existence of **.bashrc** and **.bash_profile** in root home directory using the **cd <root_home_dir>; ls -al | egrep "\.bashrc|\.bash_profile"** command.
4. If **.bash_profile** exists then:
   - Open the **<root_home_dir>/.bash_profile** file.
   - Edit the line **PATH=<path>; export PATH** where **<path>** excludes "**:**" or "**.:**" at the first part and the last part, and "**::**" or "**:.:**" at the mid part of the **root PATH** variable.
   - Save the file and go to step 7.
5. If **.bashrc** exists then:
   - Open the **<root_home_dir>/.bashrc** file.
   - Edit the line **PATH=<path>; export PATH** where **<path>** excludes "**:**" or "**.:**" at the first part and the last part, and "**::**" or "**:.:**" at the mid part of the **root PATH** variable.
   - Save the file and go to step 7.
6. If **.bash_profile** does not exist then:
   - Open the **/etc/profile** file.
   - Edit the line **PATH=<path>; export PATH** where **<path>** excludes "**:**" or "**.:**" at the first part and the last part, and "**::**" or "**:.:**" at the mid part of the **root PATH** variable.
   - Save the file and go to step 7.
7. Logout and login to the system by another session.
8. Run the **<TE_agent_root>/bin/twdaemon restart** command line to restart the TE agent.

7.1.2.51 Verify /etc/passwd File Permissions

Verify /etc/passwd File Permissions

Description	This test verifies that the 'root' user and 'root' group owns the /etc/passwd file and permissions are equal to 644 or more restrictive. Setting the recommended permissions allows users to view the file, but only 'root' has write access.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/passwd"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^-.{2}-.-{2}.-{2}.*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/passwd file. **Setting appropriate permissions and ownership on the /etc/passwd file**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -lL /etc/passwd** command. 3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/passwd** command. 4. Change ownership to **root:root** using the **chown root:root /etc/passwd** command.

7.1.2.52 World-writable Directories Should Have Their Sticky Bit Set

World-writable Directories Should Have Their Sticky Bit Set

Description	This test verifies that the 'sticky bit' is set on all world-writable directories. When the 'sticky bit' is set on a directory, only the owner of a file may remove that file from the directory.
Severity	0
Weight	5
Type	Content Test
Rules	Check Sticky Bit Setting on World Writable Files
Element	Equals "File Permissions"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /.+/ (Flags:Case insensitive) Sticky Bit Deviation Does not exist

Remediation

To remediate failure of this policy test, set sticky bit to world-writable directories.

Setting sticky bit to world-writable directories:

1. Become superuser or assume an equivalent role.
2. Run the following script:

```
PARTs=`/bin/df --local -P 2>/dev/null | /bin/awk 'NR != 1
{$1="";$2="";$3="";$4="";$5="";gsub("^[[:space:]]+/","/
",$0);print $0}' 2>/dev/null`; SaveIFS=$IFS;IFS=`/bin/echo -e
"\n\b"`; for PART in $PARTs; do /usr/bin/find "$PART" -xdev
-type d \( -perm -0002 -a ! -perm -1000 \) -ls 2>/dev/null | /bin/
awk '{a=$3; gsub("^[^/]*/","/",$0); print a, $0}'; done;
```

   to list world-writable directories which are not set the sticky bit.
3. Set the sticky bit or remove write permission for other group on directories found in step 2 using the **chmod +t <file_location>** or **chmod o-w <file_location>** command respectively.

For further details, please refer to:

http://www.redhat.com/mirrors/LDP/LDP/GNU-Linux-Tools-Summary/html/file-permissions.html

7.1.2.53 Verify Default umask for Users in /etc/profile

Verify Default umask for Users in /etc/profile

Description

This test verifies that the default umask in /etc/profile is set to 077. The umask value influences the permissions assigned to files when they are created. A misconfigured umask value could result in files with excessive permissions that can be read and/or written to by unauthorized users.

Severity

0

Weight

5

Type

Content Test

Rules

umask in /etc/profile

Excluded Nodes

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/profile"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*umask[\ \t](?![\ \t]*0*77[\ \t]*(?:$|\#)).*/ (Flags:Multiline,Comments mode)
Default umask Setting Deviation Does not exist

Remediation

To remediate failure of this policy test, set the default umask to 077 for global initialization file.

Setting the default umask to 077 for global initialization file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/profile** file.
3. Find the line that contains **umask <value>**.
4. If found, replace the **<value>** to **077**.
5. If not found, add the line **umask 077** to the file.
6. Save the file.

For further details, please run the command **man 2 umask** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/profile"
ParameterName="umask"
SeparateSymbol=" "
Value="077"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0005434
# AR_TEST_NAME = Verify Default umask for Users in /etc/profile
```

Post Remediation Category	*None*
Remediated Elements	`/etc/profile`
Post Remediation Steps	No additional Post Remediation steps

7.1.2.54 Verify Default umask for Users in /etc/bashrc

Verify Default umask for Users in /etc/bashrc

Description

This test verifies that the default umask in /etc/bashrc is set to 077. The umask value influences the permissions assigned to files when they are created. A misconfigured umask value could result in files with excessive permissions that can be read and/or written to by unauthorized users.

Severity

0

Weight

5

Type

Content Test

Rules

umask in /etc/bashrc

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/bashrc"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*umask[\ \t](?![\ \t]*0*77[\ \t]*(?:$|\#)).*/ (Flags:Multiline,Comments mode)
Default umask Setting Deviation Does not exist

Remediation

To remediate failure of this policy test, set the default umask to 077 for global initialization file.

Setting the default umask to 077 for global initialization file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/bashrc** file.
3. Find the line that contains **umask <value>**.
4. If found, replace the **<value>** to **077**.
5. If not found, add the line **umask 077** to the file.
6. Save the file.

For further details, please run the command **man 2 umask** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/bashrc"
ParameterName="umask"
SeparateSymbol=" "
Value="077"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"'"$SeparateSymbol"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0005437
# AR_TEST_NAME = Verify Default umask for Users in /etc/bashrc
```

Post Remediation Category	*None*
Remediated Elements	`/etc/bashrc`
Post Remediation Steps	No additional Post Remediation steps

7.1.2.55 Verify That /etc/cron.deny File Does Not Exist

Verify That /etc/cron.deny File Does Not Exist

Description

This test verifies that the /etc/cron.deny file does not exist.
The /etc/cron.deny file contains a list of users who are not allowed to run the 'cron' commands to submit jobs to be run at scheduled intervals. Since access to the 'cron' command is restricted using /etc/cron.allow, it is not necessary to maintain a separate deny list.

Severity

0

Weight

5

Type

Attribute Test

Rules

System Configuration Files

Element

Equals "/etc/cron.deny"

Version conditions

Action if missing:Pass
Type Does not exist

Remediation

To remediate failure of this policy test, remove the /etc/cron.deny file.

Removing the /etc/cron.deny file:

1. Become superuser or assume an equivalent role.
2. Run the **rm -rf /etc/cron.deny** command to remove the file.

For further details, please run the command **man crontab** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/cron.deny"

# Issue the command to rename the file required
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    MvLog=`/bin/mv "$FileName" "$BackupName" 2>&1`
    if [ -n "$MvLog" ]; then
        /bin/echo "FAILURE-1004: Could not delete [$FileName]
 file/directory"
        exit 1004
    else
        /bin/echo "SUCCESS-1004: [$FileName] file/directory
 deleted"
        exit 0
    fi
else
    /bin/echo "SUCCESS-1002: [$FileName] file/directory does not
 exist"
    exit 0
fi

# AR_ACTION = RHEL_FILE_DEL
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0009031
# AR_TEST_NAME = Verify That /etc/cron.deny File Does Not Exist
```

Post Remediation Category

None

Remediated Elements

None

Post Remediation Steps

No additional Post Remediation steps

7.1.2.56 Verify That /etc/at.deny File Does Not Exist

Verify That /etc/at.deny File Does Not Exist

Description

This test verifies that the /etc/at.deny file does not exist.
The /etc/at.deny file contains a list of users who are not allowed to run the 'at' commands to submit jobs to be run at scheduled intervals. Since access to the 'at' command is restricted using /etc/at.allow, it is not necessary to maintain a separate deny list.

Severity

0

Weight

5

Type

Attribute Test

Rules

System Configuration Files

Element

Equals "/etc/at.deny"

Version conditions

Action if missing:Pass
Type Does not exist

Remediation

To remediate failure of this policy test, remove the /etc/at.deny file.

Removing the /etc/at.deny file:

1. Become superuser or assume an equivalent role.
2. Run the **rm -rf /etc/at.deny** command to remove the file.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/at.deny"

# Issue the command to rename the file required
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    MvLog=`/bin/mv "$FileName" "$BackupName" 2>&1`
    if [ -n "$MvLog" ]; then
        /bin/echo "FAILURE-1004: Could not delete [$FileName]
 file/directory"
        exit 1004
    else
        /bin/echo "SUCCESS-1004: [$FileName] file/directory
 deleted"
        exit 0
    fi
else
    /bin/echo "SUCCESS-1002: [$FileName] file/directory does not
 exist"
    exit 0
fi

# AR_ACTION = RHEL_FILE_DEL
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0000811
# AR_TEST_NAME = Verify That /etc/at.deny File Does Not Exist
```

Post Remediation Category

None

Remediated Elements

None

Post Remediation Steps

No additional Post Remediation steps

7.1.2.57 Verify /etc/hosts.deny Permissions

Verify /etc/hosts.deny Permissions

Description	This test determines whether the root user owns the /etc/hosts.deny file which should be set to 644 or more restrictive permissions. This setting supports system integrity and in formation confidentiality by denying all hosts otherwise not listed in hosts.allow.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/hosts.deny"
Version conditions	Action if missing:Pass User Matches "^root\s\(\d+\)\s*$" AND Group Matches "^root\s\(\d+\)\s*$" AND Permissions Matches "^-..-.--.--.*"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/hosts.deny file. **Setting appropriate permissions and ownership on the /etc/hosts.deny file**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -lL /etc/hosts.deny** command. 3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/hosts.deny** command. 4. Change ownership to **root:root** using the **chown root:root /etc/hosts.deny** command.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-wx"
PermsRegex="-..-.--.--"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/hosts.deny"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0013665
# AR_TEST_NAME = Verify /etc/hosts.deny Permissions
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.58 Verify /etc/hosts.allow Permissions

Verify /etc/hosts.allow Permissions

Description	This test determines whether the root user owns the /etc/hosts.allow file which should be set to 644 or more restrictive permissions. Proper permissions help to prevent unautho rized modification of the file.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/hosts.allow"
Version conditions	Action if missing:Pass User Matches "^root\s\(\d+\)\s*$" AND Group Matches "^root\s\(\d+\)\s*$" AND Permissions Matches "^-..-.--.--.*"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/hosts.allow file. **Setting appropriate permissions and ownership on the /etc/hosts.allow file**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -lL /etc/hosts.allow** command. 3. Change permissions to **644** or more restrictive using the **chmod u-x,go-wx /etc/hosts.allow** command. 4. Change ownership to **root:root** using the **chown root:root /etc/hosts.allow** command.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-wx"
PermsRegex="-..-.--.--"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/hosts.allow"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0013666
# AR_TEST_NAME = Verify /etc/hosts.allow Permissions
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.59 Verify sshd_config Permissions

Verify sshd_config Permissions

Description	This test determines whether the sshd_config file is owned by the root user with permissions of 600 or more restrictive. This setting supports host integrity and information confidentiality by supporting the principle of least privilege.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	Action if missing:Pass User Matches "^root\s\(\d+\)" AND Group Matches "^root\s\(\d+\)" AND Permissions Matches "^-.{2}-{7}.*"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/ssh/sshd_config file. **Setting appropriate permissions and ownership on the /etc/ssh/sshd_config file**: 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -lL /etc/ssh/sshd_config** command. 3. Change permissions to **600** or more restrictive using the **chmod u-x,go-rwx /etc/ssh/sshd_config** command. 4. Change ownership to **root:root** using the **chown root:root /etc/ssh/sshd_config** command.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
Perms="u-x,go-rwx"
PermsRegex="-..-------"
Owner="root"
OwnersRegex="root"
Group="root"
GroupsRegex="root"
FileName="/etc/ssh/sshd_config"
ExistingElement="Pass"
FileEntry=$(/bin/ls -alLd $FileName 2>/dev/null | \
    /bin/awk '$1 ~ /^-/ {print $1,$3,$4}')

if [ -n "$FileEntry" ]; then
    if [ -n "$Owner" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$2 !~ \
            /^('$OwnersRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Owner
            OwnerLog=`(/bin/chown $Owner $FileName) 2>&1`
        fi
    fi
    if [ -n "$Group" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$3 !~ \
            /^('$GroupsRegex')$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Permissions":"$Group
            GroupLog=`(/bin/chgrp $Group $FileName) 2>&1`
        fi
    fi
    if [ -n "$Perms" ]; then
        IsInvalid=`/bin/echo "$FileEntry" | /bin/awk '$1 !~ \
            /^'$PermsRegex'$/ {print}'`
        if [ -n "$IsInvalid" ]; then
            Permissions=$Perms`[ -z "$Permissions" ] || \
                /bin/echo " "$Permissions`
            PermsLog=`(/bin/chmod $Perms $FileName) 2>&1`
        fi
    fi
    if [ -n "$PermsLog" -o -n "$OwnerLog" -o -n "$GroupLog" ];
 then
        /bin/echo "FAILURE-1005: Could not apply permissions"\
            "[$Permissions] to [$FileName] file/directory"
        exit 1005
    else
        /bin/echo "SUCCESS-1005: Permissions [$Permissions]"\
            "applied to [$FileName] file/directory"
    fi
else
    if [ "$ExistingElement" == "Pass" ]; then
        /bin/echo "SUCCESS-1002: [$FileName] file/directory does
 not exist"
    else
        /bin/echo "FAILURE-1002: [$FileName] file/directory does
 not exist"
        exit 1002
    fi
fi
exit 0

# AR_ACTION = RHEL_PERMISSIONS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0013679
# AR_TEST_NAME = Verify sshd_config Permissions
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

7.1.2.60 Verify /etc/gshadow Permissions

Verify /etc/gshadow Permissions

Description	This test verifies that the 'root' user owns /etc/gshadow and permissions are equal to 000. It is worthwhile to periodically check these file permissions as there have been package defects that changed /etc/gshadow permissions to 000.
Severity	0
Weight	5
Type	Attribute Test
Rules	System Configuration Files
Element	Equals "/etc/gshadow"
Version conditions	Action if missing:Pass User Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Group Matches "^root[\ \t]+\(\d+\)[\ \t]*$" AND Permissions Matches "^-{10}.*$"
Remediation	To remediate failure of this policy test, set appropriate permissions and ownership on the /etc/gshadow file. **Setting appropriate permissions and ownership on the /etc/gshadow file:** 1. Become superuser or assume an equivalent role. 2. Check the permissions and ownership of the file using the **ls -lL /etc/gshadow** command. 3. Change permissions to **000** using the **chmod 000 /etc/gshadow** command. 4. Change ownership to **root:root** using the **chown root:root /etc/gshadow** command.

Requirement 8 Assign a Unique ID to Each Person with Computer Access

Assigning a unique identification (ID) to each person with access ensures that each individual is uniquely accountable for their actions. When such accountability is in place, actions taken on critical data and systems are performed by, and can be traced to, known and authorized users and processes.
The effectiveness of a password is largely determined by the design and implementation of the authentication system—particularly, how frequently password attempts can be made by an attacker, and the security methods to protect user passwords at the point of entry, during transmission, and while in storage.
Note: These requirements are applicable for all accounts, including point-of-sale accounts, with administrative capabilities and all accounts used to view or access cardholder data or to access systems with cardholder data. This includes accounts used by vendors and other third parties (for example, for support or maintenance). However, Requirements 8.1.1, 8.2, 8.5, 8.2.3 through 8.2.5, and 8.1.6 through 8.1.8 are not intended to apply to user accounts within a point-of-sale payment application that only have access to one card number at a time in order to facilitate a single transaction (such as cashier accounts).

8.1 Identification Management

Define and implement policies and procedures to ensure proper user identification management for non-consumer users and administrators on all system components as follows:

8.1.1 Unique ID

Assign all users a unique ID before allowing them to access system components or cardholder data.

8.1.1.1 Reserved System Account UIDs

Reserved System Account UIDs

Description	This test verifies that UIDs 0 - 499 are reserved for system accounts. Accounts with UIDs less than 500 should include users such as root, bin, rpc, etc.
Severity	0
Weight	5
Type	Content Test
Rules	Non-System Accounts
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "Non-System Accounts"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*[^:]+:[^:]+:(?:\d|\d\d|[1-4]\d\d):/ (Flags:Multiline,Comments mode) Reserved System Account UIDs Deviation Does not exist

Remediation

To remediate failure of this policy test, update invalid UIDs (0 - 499) of non-system accounts.

Updating invalid UIDs (0 - 499) of non-system accounts:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   SYS_USER="\+|#.*|root|bin|daemon|adm|lp|sync|shutdown|halt|mail|news|uucp|operator|games|gopher|ftp|nobody|nscd|vcsa|rpc|mailnull|smmsp|pcap|ntp|dbus|avahi|sshd|rpcuser|nfsnobody|haldaemon|avahi-autoipd|distcache|apache|oprofile|webalizer|dovecot|squid|named|xfs|gdm|sabayon|abrt|dovenull|mysql|pegasus|postfix|postgres|pulse|qpidd|rtkit|saslauth|tcpdump|tomcat|usbmuxd|exim"; /bin/cat /etc/passwd 2>/dev/null 2>/dev/null | /bin/egrep -v "^[[:space:]]*($SYS_USER):" | /bin/sort -u | /bin/awk -F":" '0+$3 < 500 {print $1}'
   ```

   to list invalid accounts.
3. With invalid accounts found, run the **usermod -u <id_number> <account_name>** command to update UIDs of them to be valid (greater than **499**).

For further details, please run the command **man usermod** to read man page.

8.1.1.2 Verify No UID 0 Entries Other than root

Verify No UID 0 Entries Other than root

Description

This test verifies that the only account in /etc/passwd that has a UID of 0 is the 'root' account.
Allowing non-root accounts to have a UID of 0 would let those accounts perform actions that only 'root' should be allowed to perform.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/passwd"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /(?:^root:[^\#:&&\S]*:(?!0)\d+:|^(?!root\b)[^\#:&&\S]*:[^\#:&&\S]*:0:).*/
(Flags:Multiline,Comments mode)
Accounts with UID 0 Other than root Does not exist

Remediation

To remediate failure of this policy test, change UID of the root account to 0 and UIDs of others to not equal to 0.

Changing UID of the root account to 0 and UIDs of others to not equal to 0:

1. Become superuser or assume an equivalent role.
2. Run the **grep "^[[:space:]]*root" /etc/passwd | awk -F ":" '{print$1 " has UID equal to "$3"."}'** command to check UID of the **root** account.
3. If UID of **root** is not equal to **0**, then run the **usermod -u 0 root** command to change UID of **root** to **0**.
4. Run the **grep ":0:" /etc/passwd | grep -v "^ [[:space:]]* root" | awk -F ":" '{print$1 ":"$3":"}' | grep ":0:"** command to list all accounts (except **root**) that have UID equal to **0**.
5. Run the **usermod -u <UID> <user_name>** command to change UID of the above accounts with **<UID>** is not equal to **0**.

For further details, please run the command **man 5 passwd** to read man page.

8.1.1.3 Unique UID

Unique UID

Description

This test verifies that each user is assigned a unique UID.
Unique UIDs help prevent unauthorized access to files, processes and other system resources.

Severity

0

Weight

5

Type

Content Test

Rules

List of Duplicated UIDs

Element

Equals "Duplicated UIDs"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Duplicated UID Does not exist

Remediation

To remediate failure of this policy test, change the same UIDs of accounts.

Change the same UIDs of accounts:

1. Become superuser or assume an equivalent role.
2. Run the script:

```
DuplicatedUIDs=$(/bin/egrep  -v "^[[:space:]]*($|\#|\+)" /etc/
passwd 2>/dev/null | /bin/awk -F: '{print $3}' | /bin/sort -n | /
usr/bin/uniq -d | /bin/egrep -v "^[[:space:]]*$"); for Duplicate
dUID in $DuplicatedUIDs; do /bin/egrep  -v "^[[:space:]]*($|\#|\
+)" /etc/passwd 2>/dev/null | /bin/awk -F: '{print "UID:"$3, "Us
er:"$1}' | /bin/egrep "UID:$DuplicatedUID[[:space:]]" ; done
```

   to list all accounts having the same UID as others.
3. Run the **usermod -u <uid_value> <user_name>** command to change the same UIDs of the accounts found in step 2.

For further details, please run the command **man usermod** to read man page.

8.1.1.4 Unique Account Name

Unique Account Name

Description	This test verifies that each user is assigned a unique account name. Unique account names are useful when trying to determine which user is associated with an event, object or process.
Severity	0
Weight	5
Type	Content Test
Rules	Verify the Integrity of the System Authentication Information
Element	Equals "The System Authentication Information"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*duplicate[\ \t]+password[\ \t]+entry[\ \t]*$/ (Flags:Multiline,Comments mode) Unique Account Name Exception Does not exist
Remediation	To remediate failure of this policy test, remove duplicated user names. **Removing duplicated user names**: 1. Become superuser or assume an equivalent role. 2. Run **pwck -r /etc/passwd** command to find duplicated user names in the **/etc/passwd** file. 3. Open the **/etc/passwd** file. 4. Remove duplicated user names and save the file. For further details, please run the command **man pwck** to read man page.

8.1.1.5 Check for Duplicated Group IDs

Check for Duplicated Group IDs

Description	This test determines whether duplicate group IDs exist in the primary groups file. This setting supports system integrity by preventing a given group name from being associat ed with more than one group ID.
Severity	0
Weight	5
Type	Content Test
Rules	Duplicate GroupIDs in /etc/group
Element	Equals "Duplicate GroupIDs in /etc/group"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /.+/ (Flags:Case insensitive) Duplicate GIDs Does not exist
Remediation	To remediate failure of this policy test, remove the duplicate group IDs in the /etc/group file. **Removing the duplicate group IDs in the /etc/group file**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/group** file. 3. Find the duplicate group IDs. 4. Remove one of the duplicate group IDs and save the file. For further details, please run the command **man gpasswd** to read man page.

8.1.1.6 Check for Duplicated Group Names

Check for Duplicated Group Names

Description

This test determines whether duplicated group names exist in the primary groups file. Unique group names are useful when trying to determine which group is associated with an event, object or process.

Severity

0

Weight

5

Type

Content Test

Rules

Duplicate Group Names in /etc/group

Element

Equals "Duplicate Group Names in /etc/group"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Duplicate Group Names Does not exist

Remediation

To remediate failure of this policy test, remove the duplicate group names in the /etc/group file.

Removing the duplicate group names in the /etc/group file:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/group** file.
3. Find the duplicate group names.
4. Remove one of the duplicate group names and save the file.

For further details, please run the command **man gpasswd** to read man page.

8.1.4 Remove Inactive Users Every 90 Days

Remove/disable inactive user accounts at least every 90 days.

8.1.4.1 Verify That User Accounts Are Locked Out after 90 Days of Inactivity

Verify That User Accounts Are Locked Out after 90 Days of Inactivity

Description	This test verifies that user accounts are locked out after 90 days of inactivity. Inactive ac counts pose a threat to system security since the users are not logging in to notice failed login attempts or other anomalies.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/default/useradd"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^INACTIVE=[\ \t]*(-?\d+)$/ (Flags:Multiline,Comments mode) INACTIVE Setting Less than or equal 90 AND INACTIVE Setting Greater than or equal 0
Remediation	To remediate failure of this policy test, set the INACTIVE parameter to less than or equal to 90 and greater than 0. **Setting the INACTIVE parameter to less than or equal to 90 and greater than 0**: 1. Become superuser or assume an equivalent role. 2. Run the **useradd -D -f <value>** command to set the **INACTIVE** parameter where **<value>** is less than or equal to **90** and greater than **0**. For further details, please run the command **man useradd** to read man page.

8.1.6 Account Lockout Threshold

Limit repeated access attempts by locking out the user ID after not more than six attempts.

8.1.6.1 Limit Access Attempt to Six

Limit Access Attempt to Six

Description

This test verifies that accounts will be locked after no more than 6 failed login attempts. Locking accounts hinders the ability of an attacker to use brute-force methods to try to gain access to the system.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/password-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Element

Equals "/etc/pam.d/password-auth Content"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*auth[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_tally2\.so[\ \t]+[^\#]*\bdeny=(\d+).*/ (Flags:Multiline,Comments mode)
Failed Login Attempts Setting Greater than 0 AND
Failed Login Attempts Setting Less than or equal 6

Remediation

To remediate failure of this policy test, configure the authentication system to limit repeated access attempts by locking out the user ID after not more than six attempts.

Configuring the authentication system to limit repeated access attempts by locking out the user ID after not more than six attempts:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/password-auth** file.
3. Find the line that contains **auth <control flag>** ***[security_path/]*****pam_tally2.so** with the **<control flag>** is **required** or **requisite**.
   - If the line is found, make sure that its parameters include **deny=<value>** with the **<value>** is set to **6** or less than but not equal to **0**.
   - If the line is not found, review the **/etc/pam.d/password-auth** file and add some entries if needed to make sure that the file contains the following ordered lines:
     auth requisite ***[security_path/]*****pam_tally2.so deny=6** ***[other parameters]***
     auth sufficient ***[security_path/]*****pam_unix.so** ***[parameters]***
4. Save the file.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

8.1.7 Account Lockout Duration

Set the lockout duration to a minimum of 30 minutes or until an administrator enables the user ID.

8.1.7.1 Account Lockout Duration 30 Minutes

Account Lockout Duration 30 Minutes

Description	This test verifies that account lockout is set to at least 30 minutes. Locking accounts hinders the ability of an attacker to use brute-force methods to try to gain access to the system.
Severity	0
Weight	5
Type	Content Test
Rules	Get /etc/pam.d/password-auth Content
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 CentOS Linux release 7.2.1511
Element	Equals "/etc/pam.d/password-auth Content"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*auth[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_tally2\.so[\ \t]+[^\#&&\S\ \t]*\bunlock_time=(\S+)(?:[\ \t].*)?$/ (Flags:Multiline,Comments mode) Account Lockout Duration Greater than or equal 1800
Remediation	To remediate failure of this policy test, set the account lockout duration threshold to 1800 or greater than. **Setting the account lockout duration threshold to 1800 or greater than**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/pam.d/password-auth** file. 3. Find the line that contains **auth <control flag> *[security_path/]*pam_tally2.so** with the **<control flag>** is **required** or **requisite**. • If the line is found, make sure that its parameters include **unlock_time=<value>** with the **<value>** is set to **1800** or greater than. • If the line is not found, review the **/etc/pam.d/password-auth** file and add some entries if needed to make sure that the file contains the following ordered lines: **auth requisite *[security_path/]*pam_tally2.so deny=6 unlock_time=1800 *[other parameters]*** **auth sufficient *[security_path/]*pam_unix.so *[parameters]*** 4. Save the file. For further details, please refer to: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

8.1.8 Idle Session Timeout Threshold

If a session has been idle for more than 15 minutes, require the user to re-authenticate to re-activate the terminal or session.

8.1.8.1 Verify the idle_activation_enabled Flag Is Set to "true"

Verify the idle_activation_enabled Flag Is Set to "true"

Description

This test verifies that the idle_activation_enabled flag is set to "true". Enabling idle activation of the screen saver ensures the screensaver will be activated after the idle delay. Applications requiring continuous, real-time screen display (such as network management products) require the login session does not have administrator rights and the display station is located in a controlled-access area.

Severity

0

Weight

5

Type

Content Test

Rules

Get the idle_activation_enabled Flag Value of Gnome Screensaver

Element

Equals "Gnome Screen Saver"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*idle_activation_enabled[\ \t]+:[\ \t]+true[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode)
idle_activation_enabled Exists

Remediation

To remediate failure of this policy test, set idle_activation_enabled flag to "true".

Setting idle_activation_enabled flag to "true":

1. Become superuser or assume an equivalent role.
2. Install **GConf2** package if it is not installed:

   ```
   /bin/rpm -ivh <GConf2 package>
   ```
3. Run the following command to set idle_activation_enabled flag to "true":

   ```
   gconftool-2 --direct --config-source xml:readwrite:/etc/gconf/gconf.xml.mandatory --type bool --set /apps/gnome-screensaver/idle_activation_enabled true
   ```

For further details, please refer to:

http://projects.gnome.org/gconf/

8.1.8.2 Verify the lock_enabled Flag Is Set to "true"

Verify the lock_enabled Flag Is Set to "true"

Description	This test verifies that lock_enabled flag is set to "true". Enabling the activation of the screen lock after an idle period ensures password entry will be required in order to access the system, preventing access by passersby.
Severity	0
Weight	5
Type	Content Test
Rules	Get the lock_enabled Flag Value of Gnome Screensaver
Element	Equals "Gnome Screen Saver"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*lock_enabled[\ \t]+:[\ \t]+true[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) lock_enabled Value Exists
Remediation	To remediate failure of this policy test, set the lock_enabled flag to "true". **Setting the lock_enabled flag to "true"**: 1. Become superuser or assume an equivalent role. 2. Install **GConf2** package if it is not installed: **/bin/rpm -ivh <GConf2 package>** 3. Run the following command to set the lock_enabled flag to "true": **gconftool-2 --direct --config-source xml:readwrite:/etc/gconf/gconf.xml.mandatory --type bool --set /apps/gnome-screensaver/lock_enabled true** For further details, please refer to: http://projects.gnome.org/gconf/

8.1.8.3 Verify That ClientAliveInterval Is Set to 900 or Less and Greater than 0

Verify That ClientAliveInterval Is Set to 900 or Less and Greater than 0

Description	The two options ClientAliveInterval and ClientAliveCountMax control the timeout of ssh sessions. When the ClientAliveInterval variable is set, ssh sessions that have no activity for the specified length of time are terminated. When the ClientAliveCountMax variable is set, sshd will send client alive messages at every ClientAliveInterval interval. When the number of consecutive client alive messages are sent with no response from the client, the ssh session is terminated. It is recommended that ClientAliveInterval is set to 900 (15 minutes) or less and greater than 0.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*ClientAliveInterval[\ \t]+(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) ClientAliveInterval Timeout Less than or equal 900 AND ClientAliveInterval Timeout Greater than 0
Remediation	To remediate failure of this policy test, configure the SSH server to set a timeout interval in seconds after which if no data has been received from the client equals to 900 or less and greater than 0. **Configuring the SSH server to set a timeout interval in seconds after which if no data has been received from the client equals to 900 or less and greater than 0:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **ClientAliveInterval <value>**. 4. Set **<value>** to **900** or less and greater than **0** then save the file. 5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.

8.2 Authentication Method

In addition to assigning a unique ID, employ at least one of the following methods to authenticate all users:
- Something you know, such as a password or passphrase
- Something you have, such as a token device or smart card
- Something you are, such as a biometric

8.2.0 Authentication Method

8.2.0.1 Verify That pam_cracklib.so Has try_first-pass Option

Verify That pam_cracklib.so Has try_first-pass Option

Description

This test verifies that the system retrieve the password from a previous stacked PAM module. If not available, then prompt the user for a password.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/system-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/pam.d/system-auth_content"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\btry_first_pass\b.*$/ (Flags:Multiline,Comments mode)
try_first_pass Parameter Exists

Remediation

To remediate failure of this policy test, enable try_first_pass parameter.

Enabling try_first_pass parameter:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/system-auth** file.
3. Find the line that contains:

   password <control flag> ***[security_path/]*****pam_cracklib.so** ***[other parameters]***

   where the **<control flag>** is **required** or **requisite**.
4. If the line is found, append the **try_first_pass** parameter to the end of the line.
   - If the line is not found, review the file then edit or add some entries if needed to make sure that the file contains the following ordered lines:

     password requisite ***[security_path/]*****pam_cracklib.so** ***[other parameters]*** **try_first_pass**
     password sufficient ***[security_path/]*****pam_unix.so** ***[parameters]***
     password required ***[security_path/]*****pam_deny.so**
   - Save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html-single/Deployment_Guide/index.html#s1-pam-config-files

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System-Level_Authentication_Guide/#PAM_Configuration_Files

8.2.0.2 Verify That the System Boot Loader Is Set To Require Authentication

Verify That the System Boot Loader Is Set To Require Authentication

Description	This test verifies that the system boot loader is set to require authentication. Password protection on the boot loader configuration ensures users with physical access cannot trivially alter important bootloader settings. These include which kernel to use, and whether to enter single-user mode.
Severity	0
Weight	5
Type	Content Test
Rules	Content of grub.conf File
Excluded Nodes	Red Hat Enterprise Linux Server 6
Element	Equals "Content of grub.conf File"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*password[\ \t]+\-\-encrypted[\ \t]+\$6\$.*$/ (Flags:Multiline,Comments mode) Password protection Exists
Remediation	To remediate failure of this policy test, enable password protection to protect boot-time settings. **Enabling password protection to protect boot-time settings**: 1. Become superuser or assume an equivalent role. 2. Select a password and then generate a hash from it by running the following command: **grub-crypt --sha-512** 3. When prompted to enter a password, insert the following line into the **/etc/grub.conf** immediately after the header comments. (Use the output from **grub-crypt** as the value of ***[password-hash]***): **password --encrypted *[password-hash]*** For further details, please run the command **man grub** to read man page.

8.2.0.3 Verify Boot Loader Password Settings

Verify Boot Loader Password Settings

Description

This test verifies that a password is required when a user attempts to modify the boot process by passing commands to GRUB.
If a password is not required an attacker might be able to subvert the normal boot process on the server.

Severity

0

Weight

5

Type

Content Test

Rules

Content of grub.conf File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Content of grub.conf File"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+--md5[\ \t]+\S+/ (Flags:Multiline,Comments mode)
GRUB Password Setting Exists

Remediation

To remediate failure of this policy test, configure the grub.conf file to require a good password when a user attempts to modify the boot process by passing commands to the GRUB.

Configuring the grub.conf file to require a good password when a user attempts to modify the boot process by passing commands to the GRUB:

1. Become superuser or assume an equivalent role.
2. Run the **/sbin/grub-md5-crypt** command and type in an appropriate password to get the **md5-hashed password**.
3. Open the **/etc/grub.conf** file.
4. Add the line **password --md5 <password>** to the **/etc/grub.conf** file before the first uncommented line.
5. Replace the **<password>** with the **md5-hashed password** from step 2 and save the file.

For further details, please run the command **man grub** to read man page.

8.2.0.4 Verify That sshd_config Enables IgnoreRhosts

Verify That sshd_config Enables IgnoreRhosts

Description

This test verifies that the IgnoreRhosts setting is enabled. The use of rhosts for authentication is considered insecure.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/ssh/sshd_config"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*IgnoreRhosts[\ \t]+(\w+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode)
SSH Server IgnoreRhosts Setting Not equal "no"

Remediation

To remediate failure of this policy test, configure the SSH daemon to use safe defaults for the client and server by enabling IgnoreRhosts.

Configuring the SSH Server to enable IgnoreRhosts:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Find the line **IgnoreRhosts <value>** and set **<value>** to **yes** and save the file.
4. Run the **service sshd restart** commands to restart the **sshd** service.

For further details, please run the command **man sshd_config** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
ParameterName="IgnoreRhosts"
SeparateSymbol=" "
Value="yes"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
                "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '{IGNORECASE=1;} $1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}'
 ${FileName} \
    2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '{IGNORECASE=1;} ($1 ~ /
^[[:space:]]*'"$ParameterName"'[[:space:]]*$/) \
        {$0 = Line;}{print}'
 Line="${ParameterName}${SeparateSymbol}${Value}" \
        ${BackupName} > ${FileName}) 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]"\
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName]"\
        "parameter changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> $FileName) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line
 to"\
            "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]"\
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_CASE_INSENSITIVE
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0003254
# AR_TEST_NAME = Verify That sshd_config Enables IgnoreRhosts


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>pkill -HUP sshd</b> or <b>/
sbin/service sshd restart</b> commands to restart the <b>sshd </
b>service.</li></ol>
```

Post Remediation Category `Other`

Remediated Elements *None*

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

8.2.0.5 Require Authentication for Single User Mode

Require Authentication for Single User Mode

Description

This test verifies that single user mode requires root-level access. Authentication should always be required for root-level access. On some Linux systems single user mode is entered using the 'linux single' command in the GRUB boot-editing menu, which represents a security risk since no authentication is required.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "/etc/sysconfig/init"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*SINGLE=/sbin/sulogin[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode)
Require Authentication setting Exists

Remediation

To remediate failure of this policy test, configure the server to require authentication for single-user mode.

Configuring the init file to require authentication for single-user mode:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/sysconfig/init** file.
3. Find the line that contains **SINGLE=<value>** and set the parameter to **/sbin/sulogin**.
4. Add the line if it does not exist.
5. Save the file.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/ch-The_sysconfig_Directory.html

8.2.0.6 Verify That sshd_config Disables PermitEmptyPasswords

Verify That sshd_config Disables PermitEmptyPasswords

Description	This test verifies that the PermitEmptyPasswords option is disabled. Systems that allow users to login without passwords are extremely vulnerable to attack.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*PermitEmptyPasswords[\ \t]+(\w+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) SSH Server PermitEmptyPasswords Setting Not equal "yes"
Remediation	To remediate failure of this policy test, configure the SSH daemon to use safe defaults for the client and server by disabling PermitEmptyPasswords. **Configuring the SSH Server to disable the PermitEmptyPasswords**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **PermitEmptyPasswords <value>**. 4. Set **<value>** to **no** and save the file. 5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
ParameterName="PermitEmptyPasswords"
SeparateSymbol=" "
Value="no"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
                "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '{IGNORECASE=1;} $1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}'
 ${FileName} \
    2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '{IGNORECASE=1;} ($1 ~ /
^[[:space:]]*'"$ParameterName"'[[:space:]]*$/) \
        {$0 = Line;}{print}'
 Line="${ParameterName}${SeparateSymbol}${Value}" \
        ${BackupName} > ${FileName}) 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]"\
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName]"\
        "parameter changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> $FileName) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line
 to"\
            "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]"\
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_CASE_INSENSITIVE
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0003250
# AR_TEST_NAME = Verify That sshd_config Disables
 PermitEmptyPasswords


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>pkill -HUP sshd</b> or <b>/
sbin/service sshd restart</b> commands to restart the <b>sshd </
b>service.</li></ol>
```

Post Remediation Category

Other

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

8.2.0.7 Verify That sshd_config Disables HostbasedAuthentication

Verify That sshd_config Disables HostbasedAuthentication

Description

This test verifies that host-based authentication is disabled. Host-based authentication allows authentication to occur without any user challenge. This form of authentication is inherently insecure.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/ssh/sshd_config"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*HostbasedAuthentication[\ \t]+(\w+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode)
SSH Server HostbasedAuthentication Setting Not equal "yes"

Remediation

To remediate failure of this policy test, configure the SSH daemon to use safe defaults for the client and server by disabling HostbasedAuthentication.

Configuring the SSH Server to disable HostbasedAuthentication:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/ssh/sshd_config** file.
3. Find the line **HostbasedAuthentication <value>**.
4. Set **<value>** to **no** and save the file.
5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

For further details, please run the command **man sshd_config** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/ssh/sshd_config"
ParameterName="HostbasedAuthentication"
SeparateSymbol=" "
Value="no"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
                "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '{IGNORECASE=1;} $1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}'
 ${FileName} \
    2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
        '{IGNORECASE=1;} ($1 ~ /
^[[:space:]]*'"$ParameterName"'[[:space:]]*$/) \
        {$0 = Line;}{print}'
 Line="${ParameterName}${SeparateSymbol}${Value}" \
        ${BackupName} > ${FileName}) 2>&1`
    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]"\
            "parameter to [$Value] in [$FileName] file"
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName]"\
        "parameter changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> $FileName) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line
 to"\
            "[$FileName] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]"\
        "line added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_CASE_INSENSITIVE
# AR_COMPLETION = COMPLETION_OTHER
# AR_TEST_ID = T0003249
# AR_TEST_NAME = Verify That sshd_config Disables
 HostbasedAuthentication


# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>pkill -HUP sshd</b> or <b>/
sbin/service sshd restart</b> commands to restart the <b>sshd </
b>service.</li></ol>
```

Post Remediation Category

```
Other
```

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service.

8.2.0.8 Verify That retry Option Is Set to 3 or Less

Verify That retry Option Is Set to 3 or Less

Description

This test verifies that the system is configured to allow 3 tries before sending back a failure.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/system-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/pam.d/system-auth_content"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\bretry=[1-3]\b.*$|^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t](?![^\#\n]*\bretry=).*$/ (Flags:Multiline,Comments mode)
retry Setting Exists

Remediation

To remediate failure of this policy test, set the retry option to less than or equal to 3.

Setting the retry option to less than or equal to 3:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/system-auth** file.
3. Find the line that contains:

   password <control flag> ***[security_path/]*****pam_cracklib.so** ***[other parameters]***

   where the **<control flag>** is **required** or **requisite**.
4. If the line is found, set the **retry** option to less than or equal to **3**.
   - If the line is not found, review the file then edit or add some entries if needed to make sure that the file contains the following ordered lines:

     password requisite ***[security_path/]*****pam_cracklib.so** ***[other parameters]*** **retry=<value>**
     password sufficient ***[security_path/]*****pam_unix.so** ***[parameters]***
     password required ***[security_path/]*****pam_deny.so**

     where **<value>** is less than or equal to **3**.
   - Save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html-single/Deployment_Guide/index.html#s1-pam-config-files

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System-Level_Authentication_Guide/#PAM_Configuration_Files

8.2.3 Password Length and Complexity

Passwords/phrases must meet the following:
- Require a minimum length of at least seven characters.
- Contain both numeric and alphabetic characters.
Alternatively, the passwords/phrases must have complexity and strength at least equivalent to the parameters specified above.

8.2.3.1 Password Length

Require a minimum password length of at least seven characters.

8.2.3.1.1 Password Length

Password Length

<table>
<tr><th></th><th></th></tr>
<tr><td><b>Description</b></td><td>This test verifies that the system is configured to use a minimum password length of 7 characters.
Using longer passwords hinders the ability of an attacker to use brute-force methods to try to gain access to the system.</td></tr>
<tr><td><b>Severity</b></td><td>0</td></tr>
<tr><td><b>Weight</b></td><td>5</td></tr>
<tr><td><b>Type</b></td><td>Content Test</td></tr>
<tr><td><b>Rules</b></td><td>Get Password Modules Configured in /etc/pam.d/passwd File</td></tr>
<tr><td><b>Excluded Nodes</b></td><td>Red Hat Enterprise Linux Server 7
Red Hat Enterprise Linux Server 6</td></tr>
<tr><td><b>Element</b></td><td>Equals "Password Configuration"</td></tr>
<tr><td><b>Version conditions</b></td><td>If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\bminlen=(\d+)\b.*$/ (Flags:Multiline,Comments mode)
Minimum of Password Length Setting Greater than or equal 7</td></tr>
</table>

Remediation

To remediate failure of this policy test, set the required minimum acceptable size for the new password to at least 7 characters.

Setting the minimum acceptable size for the new password to at least 7 characters:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   directory="/etc/pam.d"; files="passwd;"; files=$files$(/bin/cat /etc/pam.d/passwd 2>/dev/null | /bin/awk -F"#" ' $0 ~ /^[[:space:]]*password[[:space:]]+include|substack[[:space:]].*/ {print $1}' | /bin/awk 'BEGIN {ORS=";"} {print $3}'); SavedIFS=$IFS; IFS=";"; for file in $files; do if [ "`/usr/bin/dirname $file 2>/dev/null`" != "." ]; then if [ -f "$file" ]; then /bin/echo $file; fi; else full_path=$directory"/"$file; if [ -f "$full_path" ]; then /bin/echo $full_path; fi; fi; done; IFS=$SavedIFS;
   ```

   to list the paths of the PAM configuration files need to update.
3. For each file listed in step 2, open it.
   - Find the line that contains:

     password <control_flag> *[security_path/]*pam_cracklib.so *[parameters]*

     where the **<control flag>** is **requisite** or **required**.
   - If the line is found, find the parameter **minlen=<value>**:
     - If the parameter is found, set the **<value>** to **7** or greater.
     - If the parameter is not found, add the parameter **minlen=<value>** to the line where the **<value>** is set to **7** or greater.
4. If the line is not found in any file listed in step 2, then:
   - Review the **/etc/pam.d/passwd** file, then add one entry if needed to make sure it contains the line:

     password include system-auth
     or
     password substack system-auth
   - Review the **/etc/pam.d/system-auth** file, then edit or add some entries if needed to make sure that the file contains the following ordered lines:

     password requisite *[security_path/]*pam_cracklib.so minlen=<value> *[other parameters]*
     password sufficient *[security_path/]*pam_unix.so use_authtok *[other parameters]*
     password required *[security_path/]*pam_deny.so

     where the **<value>** is set to **7** or greater.
5. Save the file.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

8.2.3.2 Password Complexity

Passwords must contain both numeric and alphabetic characters.

8.2.3.2.1 Verify That Passwords Contain No More than 3 Consecutive Repeating Characters

Verify That Passwords Contain No More than 3 Consecutive Repeating Characters

Description

This test verifies that the system must require passwords contain no more than three consecutive repeating characters.
To enforce the use of complex passwords, the number of consecutive repeating characters is limited. Passwords with excessive repeated characters may be more vulnerable to password-guessing attacks.

Severity

0

Weight

5

Type

Content Test

Rules

Get Password Modules Configured in /etc/pam.d/passwd File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Password Configuration"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#]*\bmaxrepeat=(\d+)\b.*$/ (Flags:Multiline,Comments mode)
The Number of Consecutive Repeating Characters Greater than 0 AND
The Number of Consecutive Repeating Characters Less than or equal 3

Remediation

To remediate failure of this policy test, set the required passwords contain no more than three consecutive repeating characters.

Setting the required passwords contain no more than three consecutive repeating characters:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   directory="/etc/pam.d"; files="passwd; "; files=$files$(/bin/cat /etc/pam.d/passwd 2>/dev/null | /bin/awk -F"#" ' $0 ~ /^[[:space:]]*password[[:space:]]+include|substack[[:space:]].*/ { print $1 }' | /bin/awk 'BEGIN { ORS="; " } { print $3 }'); SavedIFS=$IFS; IFS="; "; for file in $files; do if [ "`/usr/bin/dirname $file 2>/dev/null`" != "." ]; then if [ -f "$file" ]; then /bin/echo "$file"; fi; else full_path=$directory"/"$file; if [ -f "$full_path" ]; then /bin/echo "$full_path"; fi; fi; done; IFS=$SavedIFS;
   ```

   to list the paths of the PAM configuration files need to update.
3. For each file listed in step 2, open it.
   - Find the line that contains:

     password <control_flag> *[security_path/]*pam_cracklib.so *[parameters]*

     where the **<control flag>** is **requisite** or **required**.
   - If the line is found, find the parameter **maxrepeat =<value>**:
     - If the parameter is found, set the **<value>** to **3** or less but greater than **0**.
     - If the parameter is not found, add the parameter **maxrepeat =<value>** to the line where the **<value>** is set to **3** or less but greater than **0**.
4. If the line is not found in any file listed in step 2, then:
   - Review the **/etc/pam.d/passwd** file, then add one entry if needed to make sure it contains the line:

     password required *[security_path/]*pam_stack.so service=system-auth
     or
     password include|substack system-auth
   - Review the **/etc/pam.d/system-auth** file, then edit or add some entries if needed to make sure that the file contains the following ordered lines:

     password requisite *[security_path/]*pam_cracklib.so maxrepeat =<value> *[other parameters]*
     password sufficient *[security_path/]*pam_unix.so use_authtok *[other parameters]*
     password required *[security_path/]*pam_deny.so

     where the **<value>** is set to **3** or less but greater than **0**.
5. Save the file.

For further details, please refer to:

https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

8.2.3.2.2 Password Character Mix: At Least a Numerical Character

Password Character Mix: At Least a Numerical Character

Description

This test verifies that passwords include at least a numerical character.Forcing users to use complex passwords makes it more difficult for attackers to gain access to the system.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/system-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/pam.d/system-auth_content"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\bdcredit=-(\d+)\b.*/ (Flags:Multiline,Comments mode)
Minimum of Numerical Password Characters Greater than or equal 1

Remediation

To remediate failure of this policy test, set the required minimum number of digits to at least 1.

Setting the required minimum digits to at least 1:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/system-auth** file.
3. Find the line that contains:

   password <control flag> ***[security_path/]*****pam_cracklib.so** ***[other parameters]***

   where the **<control flag>** is **required** or **requisite**.
4. If the line is found, find the parameter **dcredit=<value>**:
   - If the parameter is found, then change the **<value>** to **-1** or less.
   - If the parameter is not found, then add the **dcredit=<value>** parameter to the line where the **<value>** is set to **-1** or less.
5. If the line is not found, review the file then edit or add some entries if needed to make sure that the file contains the following ordered lines:

   password requisite ***[security_path/]*****pam_cracklib.so dcredit=<value>** ***[other parameters]***
   password sufficient ***[security_path/]*****pam_unix.so use_authtok** ***[other parameters]***
   password required ***[security_path/]*****pam_deny.so**

   where the **<value>** is set to **-1** or less.
6. Save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html-single/Deployment_Guide/index.html#s1-pam-config-files

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System-Level_Authentication_Guide/#PAM_Configuration_Files

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/pam.d/system-auth"
Parameter="dcredit"
Value="-1"
Module="pam_cracklib.so"
Regex="^[[:space:]]*password[[:space:]]+(requisite|required)
[[:space:]]+([^\#]+\/)?pam_cracklib\.so"
ParameterRegex="\<${Parameter}=-?[0-9]+\>"

ExistedPamCrackLib=`/bin/egrep "${Regex}" $FileName 2>/dev/null`
if [ -z "$ExistedPamCrackLib" ]; then
    /bin/echo "FAILURE-7001: [$Module] module is not plugged
 into"\
        "the [$FileName] file"
    exit 7001
fi
# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

IsExisted=`(/bin/echo $ExistedPamCrackLib | /bin/awk -F"#"
 '{print $1}' | /bin/egrep "${ParameterRegex}") 2>&1`
# Issue the command to change a field
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"#" 'BEGIN{OFS="#"}
    $1 ~ /'$Regex'/ {
    gsub(/'$ParameterRegex'/,"'$Parameter'='$Value'",$1)
    }{print}' "$BackupName" > "$FileName") 2>&1`
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-7001: Could not change value of
 [$Parameter] field"\
            "to [$Value] in [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: Value of [$Parameter] field"\
        "changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/awk -F"#" '$0 ~ /'"$Regex"'/\
        {if (NF == 1) $0 = $1" '"$Parameter=$Value"'";
        else $0 = $1" '"$Parameter=$Value"'#"$2;} {print}' \
        ${BackupName} > ${FileName}) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-7001: Could not add [$Parameter=
$Value] field"\
            "to [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: [$Parameter=$Value] field"\
        "added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0019065
# AR_TEST_NAME = Password Character Mix: At Least A Numerical
 Character
```

Post Remediation Category

None

Remediated Elements

```
/etc/pam.d/system-auth
/etc/pam.d/system-auth-ac
```

Post Remediation Steps

No additional Post Remediation steps

8.2.3.2.3 Password Character Mix: At Least an Uppercase Character

Password Character Mix: At Least an Uppercase Character

Description

This test verifies that passwords include at least an uppercase alphabetic character.Forcing users to use complex passwords makes it more difficult for attackers to gain access to the system.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/system-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/pam.d/system-auth_content"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\bucredit=-(\d+)\b.*/ (Flags:Multiline,Comments mode)
Minimum of Uppercase Password Characters Greater than or equal 1

Remediation

To remediate failure of this policy test, set the required minimum number of upper case characters to at least 1.

Setting the required minimum number of upper case characters to at least 1:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/system-auth** file.
3. Find the line that contains:

   password <control flag> ***[security_path/]*****pam_cracklib.so** ***[other parameters]***

   where the **<control flag>** is **required** or **requisite**.
4. If the line is found, find the parameter **ucredit=<value>**:
   - If the parameter is found, then change the **<value>** to **-1** or less.
   - If the parameter is not found, then add the **ucredit=<value>** parameter to the line where the **<value>** is set to **-1** or less.
5. If the line is not found, review the file then edit or add some entries if needed to make sure that the file contains the following ordered lines:

   password requisite ***[security_path/]*****pam_cracklib.so ucredit=<value>** ***[other parameters]***
   password sufficient ***[security_path/]*****pam_unix.so use_authtok** ***[other parameters]***
   password required ***[security_path/]*****pam_deny.so**

   where the **<value>** is set to **-1** or less.
6. Save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-pam-sample-simple.html

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html-single/Deployment_Guide/index.html#s1-pam-config-files

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System-Level_Authentication_Guide/#PAM_Configuration_Files

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/pam.d/system-auth"
Parameter="ucredit"
Value="-1"
Module="pam_cracklib.so"
Regex="^[[:space:]]*password[[:space:]]+(requisite|required)
[[:space:]]+([^\#]+\/)?pam_cracklib\.so"
ParameterRegex="\<${Parameter}=-?[0-9]+\>"

ExistedPamCrackLib=`/bin/egrep "${Regex}" $FileName 2>/dev/null`
if [ -z "$ExistedPamCrackLib" ]; then
    /bin/echo "FAILURE-7001: [$Module] module is not plugged
 into"\
        "the [$FileName] file"
    exit 7001
fi
# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

IsExisted=`(/bin/echo $ExistedPamCrackLib | /bin/awk -F"#"
 '{print $1}' | /bin/egrep "${ParameterRegex}") 2>&1`
# Issue the command to change a field
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"#" 'BEGIN{OFS="#"}
    $1 ~ /'$Regex'/ {
    gsub(/'$ParameterRegex'/,"'$Parameter'='$Value'",$1)
    }{print}' "$BackupName" > "$FileName") 2>&1`
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-7001: Could not change value of
 [$Parameter] field"\
            "to [$Value] in [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: Value of [$Parameter] field"\
        "changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/awk -F"#" '$0 ~ /'"$Regex"'/\
        {if (NF == 1) $0 = $1" '"$Parameter=$Value"'";
        else $0 = $1" '"$Parameter=$Value"'#"$2;} {print}' \
        ${BackupName} > ${FileName}) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-7001: Could not add [$Parameter=
$Value] field"\
            "to [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: [$Parameter=$Value] field"\
        "added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0019064
# AR_TEST_NAME = Password Character Mix: At Least An Uppercase
 Character
```

Post Remediation Category	*None*
Remediated Elements	`/etc/pam.d/system-auth` `/etc/pam.d/system-auth-ac`
Post Remediation Steps	No additional Post Remediation steps

8.2.3.2.4 Password Character Mix: At Least a Lowercase Character

Password Character Mix: At Least a Lowercase Character

Description

This test verifies that passwords include at least a lowercase alphabetic character. Forcing users to use complex passwords makes it more difficult for attackers to gain access to the system.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/system-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/pam.d/system-auth_content"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\blcredit=-(\d+)\b.*/ (Flags:Multiline,Comments mode)
Minimum of Lowercase Password Characters Greater than or equal 1

Remediation

To remediate failure of this policy test, set the required minimum number of lower case characters to at least 1.

Setting the required minimum number of lower case characters to at least 1:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/system-auth** file.
3. Find the line that contains:

   password <control flag> ***[security_path/]*****pam_cracklib.so** ***[other parameters]***

   where the **<control flag>** is **required** or **requisite**.
4. If the line is found, find the parameter **lcredit=<value>**:
   - If the parameter is found, then change the **<value>** to **-1** or less.
   - If the parameter is not found, then add the **lcredit=<value>** parameter to the line where the **<value>** is set to **-1** or less.
5. If the line is not found, review the file then edit or add some entries if needed to make sure that the file contains the following ordered lines:

   password requisite ***[security_path/]*****pam_cracklib.so lcredit=<value>** ***[other parameters]***
   password sufficient ***[security_path/]*****pam_unix.so use_authtok** ***[other parameters]***
   password required ***[security_path/]*****pam_deny.so**

   where the **<value>** is set to **-1** or less.
6. Save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-pam-sample-simple.html

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html-single/Deployment_Guide/index.html#s1-pam-config-files

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System-Level_Authentication_Guide/#PAM_Configuration_Files

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/pam.d/system-auth"
Parameter="lcredit"
Value="-1"
Module="pam_cracklib.so"
Regex="^[[:space:]]*password[[:space:]]+(requisite|required)
[[:space:]]+([^\#]+\/)?pam_cracklib\.so"
ParameterRegex="\<${Parameter}=-?[0-9]+\>"

ExistedPamCrackLib=`/bin/egrep "${Regex}" $FileName 2>/dev/null`
if [ -z "$ExistedPamCrackLib" ]; then
    /bin/echo "FAILURE-7001: [$Module] module is not plugged
 into"\
        "the [$FileName] file"
    exit 7001
fi
# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

IsExisted=`(/bin/echo $ExistedPamCrackLib | /bin/awk -F"#"
 '{print $1}' | /bin/egrep "${ParameterRegex}") 2>&1`
# Issue the command to change a field
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"#" 'BEGIN{OFS="#"}
    $1 ~ /'$Regex'/ {
    gsub(/'$ParameterRegex'/,"'$Parameter'='$Value'",$1)
    }{print}' "$BackupName" > "$FileName") 2>&1`
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-7001: Could not change value of
 [$Parameter] field"\
            "to [$Value] in [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: Value of [$Parameter] field"\
        "changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/awk -F"#" '$0 ~ /'"$Regex"'/\
        {if (NF == 1) $0 = $1" '"$Parameter=$Value"'";
        else $0 = $1" '"$Parameter=$Value"'#"$2;} {print}' \
        ${BackupName} > ${FileName}) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-7001: Could not add [$Parameter=
$Value] field"\
            "to [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: [$Parameter=$Value] field"\
        "added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0019063
# AR_TEST_NAME = Password Character Mix: At Least a Lowercase
 Character
```

Post Remediation Category *None*

Remediated Elements

```
/etc/pam.d/system-auth
/etc/pam.d/system-auth-ac
```

Post Remediation Steps No additional Post Remediation steps

8.2.3.2.5 Verify That Minimum Special Password Characters Setting in the /etc/pam.d/system-auth File Is Greater than or Equal to 1

Verify That Minimum Special Password Characters Setting in the /etc/pam.d/system-auth File Is Greater than or Equal to 1

Description

This test verifies that passwords include at least a special character. Forcing users to use complex passwords makes it more difficult for attackers to gain access to the system.

Severity

0

Weight

5

Type

Content Test

Rules

Get /etc/pam.d/system-auth Content

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/pam.d/system-auth_content"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#\n]*\bocredit=-(\d+)\b.*/ (Flags:Multiline,Comments mode)
Minimum of Special Password Characters Greater than or equal 1

Remediation

To remediate failure of this policy test, set the required minimum number of special characters to at least 1.

Setting the required minimum number of special characters to at least 1:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/pam.d/system-auth** file.
3. Find the line that contains:

   password <control flag> ***[security_path/]*****pam_cracklib.so** ***[other parameters]***

   where the **<control flag>** is **required** or **requisite**.
4. If the line is found, find the parameter **ocredit=<value>**:
   - If the parameter is found, then change the **<value>** to **-1** or less.
   - If the parameter is not found, then add the **ocredit=<value>** parameter to the line where the **<value>** is set to **-1** or less.
5. If the line is not found, review the file then edit or add some entries if needed to make sure that the file contains the following ordered lines:

   password requisite ***[security_path/]*****pam_cracklib.so ocredit=<value>** ***[other parameters]***
   password sufficient ***[security_path/]*****pam_unix.so use_authtok** ***[other parameters]***
   password required ***[security_path/]*****pam_deny.so**

   where the **<value>** is set to **-1** or less.
6. Save the file.

For further details, please refer to:

RHEL 5:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-pam-sample-simple.html

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html-single/Deployment_Guide/index.html#s1-pam-config-files

RHEL 6:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

RHEL 7:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System-Level_Authentication_Guide/#PAM_Configuration_Files

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/pam.d/system-auth"
Parameter="ocredit"
Value="-1"
Module="pam_cracklib.so"
Regex="^[[:space:]]*password[[:space:]]+(requisite|required)
[[:space:]]+([^\#]+\/)?pam_cracklib\.so"
ParameterRegex="\<${Parameter}=-?[0-9]+\>"

ExistedPamCrackLib=`/bin/egrep "${Regex}" $FileName 2>/dev/null`
if [ -z "$ExistedPamCrackLib" ]; then
    /bin/echo "FAILURE-7001: [$Module] module is not plugged
 into"\
        "the [$FileName] file"
    exit 7001
fi
# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

IsExisted=`(/bin/echo $ExistedPamCrackLib | /bin/awk -F"#"
 '{print $1}' | /bin/egrep "${ParameterRegex}") 2>&1`
# Issue the command to change a field
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"#" 'BEGIN{OFS="#"}
    $1 ~ /'$Regex'/ {
    gsub(/'$ParameterRegex'/,"'$Parameter'='$Value'",$1)
    }{print}' "$BackupName" > "$FileName") 2>&1`
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-7001: Could not change value of
 [$Parameter] field"\
            "to [$Value] in [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: Value of [$Parameter] field"\
        "changed to [$Value] in [$FileName] file"
else
    AddLog=`(/bin/awk -F"#" '$0 ~ /'"$Regex"'/\
        {if (NF == 1) $0 = $1" '"$Parameter=$Value"'";
        else $0 = $1" '"$Parameter=$Value"'#"$2;} {print}' \
        ${BackupName} > ${FileName}) 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-7001: Could not add [$Parameter=
$Value] field"\
            "to [$FileName] file"
        # Rollback to the original file
        /bin/cp -f ${BackupName} $FileName 2>/dev/null
        exit 7001
    fi
    /bin/echo "SUCCESS-7001: [$Parameter=$Value] field"\
        "added to [$FileName] file"
fi
exit 0

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0019066
# AR_TEST_NAME = Verify That Minimum Special Password Characters
 Setting in the /etc/pam.d/system-auth File Is Greater than or
 Equal to 1
```

Post Remediation Category

None

Remediated Elements

```
/etc/pam.d/system-auth
/etc/pam.d/system-auth-ac
```

Post Remediation Steps

No additional Post Remediation steps

8.2.4 Password Aging

Change user passwords / passphrases at least every 90 days.

8.2.4.1 Verify PASS_MAX_DAYS Parameter in /etc/login.defs

Verify PASS_MAX_DAYS Parameter in /etc/login.defs

Description	This test verifies that /etc/login.defs is configured to force password change after 90 days or less. This setting is used for the creation of new accounts. Requiring regular password changes ensures that if a password is cracked, it will only be valid temporarily.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/login.defs"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*PASS_MAX_DAYS[\ \t]+(\d+)[\ \t]*(?:$|\#)/ (Flags:Multiline,Comments mode) PASS_MAX_DAYS Less than or equal 90 AND PASS_MAX_DAYS Greater than 0
Remediation	To remediate failure of this policy test, set the maximum number of days a password may be used to at least 1, but not greater than 90. **Setting the maximum number of days a password may be used to at least 1, but not greater than 90**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/login.defs** file. 3. Find the line **PASS_MAX_DAYS <value>**. 4. Set the **<value>** to greater than **0** and less than or equal **90** and save the file. For further details, please run the command **man login.defs** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/login.defs"
ParameterName="PASS_MAX_DAYS"
SeparateSymbol=" "
Value="90"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"'"'"$SeparateSymbol"'"'"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0003381
# AR_TEST_NAME = Verify PASS_MAX_DAYS Parameter in /etc/
login.defs
```

Post Remediation Category	*None*
Remediated Elements	*None*
Post Remediation Steps	No additional Post Remediation steps

8.2.4.2 Verify PASS_MAX_DAYS Setting for Non-system Accounts

Verify PASS_MAX_DAYS Setting for Non-system Accounts

Description	This test verifies that all non-system accounts are configured to expire every 90 days or less. Requiring regular password changes ensures that if a password is cracked, it will only be valid temporarily.
Severity	0
Weight	5
Type	Content Test
Rules	Verify Expiration Password for Non-system Account
Excluded Nodes	Oracle Linux Server release 5.8 CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 Oracle Linux Server release 5.10 Red Hat Enterprise Linux Server 6 CentOS 6 Oracle Linux Server release 5.11 CentOS Linux release 7.2.1511 Red Hat Enterprise Linux Server 5 CentOS 5 Oracle Linux Server release 5.9
Element	Equals "Expiration Password"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^\S+:.*PASS_MAX_DAYS=(?!0*[1-8][0-9]?\b|0*90?\b).*/ (Flags:Multiline,Comments mode) 'Fail Maximum Password Age' for Non-system Accounts Does not exist
Remediation	To remediate failure of this policy test, set the maximum number of days during which a password is valid to 90 or less for non-system accounts. **Setting the maximum number of days during which a password is valid to 90 or less for non-system accounts**: 1. Become superuser or assume an equivalent role. 2. Run the script: **for Acc in `awk -F: '$1 !~ /^[[:space:]]*#/ && $3>=500 && $3!=65534 {print $1}' /etc/passwd 2>/dev/null`; do awk -F: '$1 ~ /^[[:space:]]*'$Acc'$/ && $2!~/[!*]+/ && ($5>90 || $5 ~ /^[[:space:]]*$/ || $5 == 0) {print $1":PASS_MAX_DAYS="$5}' /etc/shadow 2>/dev/null; done** to list non-system accounts of which the maximum number of days during which a password is valid is greater than **90**. 3. Change the maximum number of days during which a password is valid to **90** or less for non-system accounts found in step 2 using the **chage -M <value> <user_name>** command, where **<value>** is less than or equal to **90**. For further details, please run the command **man chage** to read man page.
Command Line	/bin/sh $(ScriptFile.sh)

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
PasswordParameter="PASS_MAX_DAYS"
Value="90"
FailedAccounts=`/bin/awk -F":" '$1 !~ /[[:space:]]*#/ && $2!~/[!
*]+/ {
    GetIdCmd="/usr/bin/id -u " $1 " 2>/dev/null"; Uid=""
    GetIdCmd | getline Uid
    if(Uid ~ /^[0-9]+$/ && 0+Uid >= 500 && 0+Uid < 65534){
        if($5 !~ /^-?[0-9]+$/ || 0+$5 > 90){ print $1 }
    }
}' /etc/shadow 2>/dev/null`

# Issue the command to change PASS_MAX_DAYS setting for non-
system accounts
SavedIFS=$IFS
IFS=`/bin/echo -ne "\n\b"`

if [ -n "${FailedAccounts}" ]; then
    for Account in $FailedAccounts; do
        UpdateLog=`/usr/bin/chage -M $Value $Account 2>&1`
        if [ -n "$UpdateLog" ]; then
            FailureUpdate=`[ -z "$FailureUpdate" ] ||\
                /bin/echo $FailureUpdate"\n"`$Account
        else
            SuccessUpdate=`[ -z "$SuccessUpdate" ] ||\
                /bin/echo $SuccessUpdate"\n"`$Account
        fi
    done
else
    /bin/echo "SUCCESS-7001: No account with failure
 [$PasswordParameter]"
    exit 0
fi
IFS=$SavedIFS

if [ -n "${FailureUpdate}" ]; then
    /bin/echo -e "FAILURE-7001: Could not change
 [$PasswordParameter]"\
        "to [$Value] for [$FailureUpdate] account"
    if [ -n "${SuccessUpdate}" ]; then
        /bin/echo -e "Changed [$PasswordParameter]"\
            "to [$Value] for [$SuccessUpdate] account"
    fi
    exit 7001
else
    /bin/echo -e "SUCCESS-7001: Changed [$PasswordParameter]"\
        "to [$Value] for [$SuccessUpdate] account"
    exit 0
fi

# AR_ACTION = RHEL_OTHERS
# AR_COMPLETION = COMPLETION_NONE
# AR_TEST_ID = T0006757
# AR_TEST_NAME = Verify PASS_MAX_DAYS Setting for Non-system
 Accounts
```

Post Remediation Category

None

Remediated Elements

```
/etc/shadow
/etc/shadow-
```

Post Remediation Steps

No additional Post Remediation steps

8.2.5 Password History

Do not allow an individual to submit a new password that is the same as any of the last four passwords he or she has used.

8.2.5.1 Verify That the difok Parameter Is Set to 4 or More

Verify That the difok Parameter Is Set to 4 or More

Description

This test verifies that the system must require a password containing the minimum number of characters that must be different from the previous password at least 4.
To ensure password changes are effective in their goals, the system must ensure that old and new passwords have significant differences. Without significant changes, new passwords may be easily guessed based on the value of a previously compromised password.

Severity

0

Weight

5

Type

Content Test

Rules

Get Password Modules Configured in /etc/pam.d/passwd File

Excluded Nodes

Red Hat Enterprise Linux Server 6

Element

Equals "Password Configuration"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+(?:requisite|required)[\ \t]+[^\#&&\S]*\bpam_cracklib\.so[\ \t]+[^\#]*\bdifok=(\d+)\b.*/ (Flags:Multiline,Comments mode)
difok Parameter Greater than or equal 4

Remediation

To remediate failure of this policy test, set the required minimum number of characters that must be different from the previous password at least 4.

Setting the required minimum number of characters that must be different from the previous password at least 4:

1. Become superuser or assume an equivalent role.
2. Run the script:

```
directory="/etc/pam.d"; files="passwd;"; files=$files$(/bin/
cat /etc/pam.d/passwd 2>/dev/null | /bin/awk -F"#" ' $0 ~ /
^[[:space:]]*password[[:space:]]+include|substack[[:spac
e:]].*/ {print $1}' | /bin/awk 'BEGIN {ORS=";"} {print $3}');
SavedIFS=$IFS; IFS=";"; for file in $files; do if [ "`/usr/bin/
dirname $file 2>/dev/null`" != "." ]; then if [ -f "$file" ]; then /
bin/echo $file; fi; else full_path=$directory"/"$file; if [ -f "$ful
l_path" ]; then /bin/echo $full_path; fi; fi; done; IFS=$Saved
IFS;
```

   to list the paths of the PAM configuration files need to update.

3. For each file listed in step 2, find the line that contains:

   password <control_flag> ***[security_path/]*****pam_cracklib.so**
   [parameters]

   where the **<control flag>** is **requisite** or **required**.
4. If the line is found, find the parameter **difok=<value>**:
   - If the parameter is found, set the **<value>** to **4** or greater.
   - If the parameter is not found, add the parameter **difok=<value>** to the line where the **<value>** is set to **4** or greater.
5. If the line is not found in any file listed in step 2, then:
   - Review the **/etc/pam.d/passwd** file, then add one entry if needed to make sure it contains the line:

     password include system-auth
     or
     password substack system-auth
   - Review the **/etc/pam.d/system-auth** file, then edit or add some entries if needed to make sure that the file contains the following ordered lines:

     password requisite ***[security_path/]*****pam_cracklib.so**
     difok=<value> ***[other parameters]***
     password sufficient ***[security_path/]*****pam_unix.so**
     use_authtok ***[other parameters]***
     password required ***[security_path/]*****pam_deny.so**

     where the **<value>** is set to **4** or greater.
6. Save the file.

For further details, please refer to:

http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Smart_Cards/PAM_Configuration_Files.html

8.2.5.2 Password Reuse

Password Reuse

Description

This test verifies that passwords cannot be reused until at least 4 changes have been made.
Preventing users from reusing passwords makes it more difficult for attackers to gain access to the system.

Severity

0

Weight

5

Type

Content Test

Rules

Verify Password Reuse Setting in /etc/pam.d/system-auth File

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "Password_Reuse"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*password[\ \t]+[^\#\n]*\bpam_unix\.so[\ \t]+[^\#\n]*\bremember=(?:[^0-3]|\d{2,})\b.*$|^[\ \t]*password[\ \t]+[^\#\n]*\bpam_pwhistory\.so\b(?![^\#\n]*\bremember=[0-3]\b).*$/ (Flags:Multiline,Comments mode)
Password Reuse Setting Exists

Remediation

To remediate failure of this policy test, set the 'remember' option for pam_unix.so (or pam_pwhistory.so in RHEL 5) and create /etc/security/opasswd file in order to prevent users from reusing the last 4 old passwords.

Setting the 'remember' option and creating /etc/security/opasswd file in order to prevent users from reusing the last 4 old passwords:

1. Become superuser or assume an equivalent role.
2. Run the **touch /etc/security/opasswd** command to create the **/etc/security/opasswd** file if it does not exist.
3. Open the **/etc/pam.d/system-auth** file.
4. Find the line that contains **password <control_flag>*[security_path/]*pam_unix.so *[parameters]*** or **password <control_flag>*[security_path/]*pam_pwhistory.so *[parameters]***.
   - If one of the lines is found, then find **remember=<value>** parameter.
     - If the parameter is found, set the **<value>** to **4** or greater.
     - If the parameter is not found, add the parameter **remember=<value>** to the line where **<value>** is set to **4** or greater.
   - If none of the lines is found, review the **/etc/pam.d/system-auth** file, then edit or add some entries if needed to make sure that the file contains the following ordered lines:

     password sufficient *[security_path/]*pam_unix.so remember=<value> *[other parameters]*
     or
     password required *[security_path/]*pam_pwhistory.so remember=<value> *[other parameters]*
     password sufficient *[security_path/]*pam_unix.so *[parameters]*

     where the **<value>** is set to **4** or greater.
5. Save the file.

For further details, please refer to:

https://access.redhat.com/site/documentation//en-US/Red_Hat_Enterprise_Linux/6/html-single/Managing_Smart_Cards/index.html#Pluggable_Authentication_Modules

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
OpasswdFile="/etc/security/opasswd"
Parameter="remember"
Value="4"
ParameterRegex="\<$Parameter[[:space:]]*=([[:space:]]*-?[0-9]*|
\w*)"
# Check if current OS is RHEL 4 or RHEL 5
Version=`/bin/cat /etc/redhat-release 2>/dev/null | \
    /bin/awk -F"release" '{print $2}' | /bin/awk -F"(" '{print
 $1}' | \
    /bin/awk -F"." '{print $1}' | /bin/sed -e 's/ //g'`;

FileNames="/etc/pam.d/system-auth"

if [ "$Version" = "5" ]; then
    Regex="^[[:space:]]*password[[:space:]]+[^\#]+"
    Regex=$Regex"[[:space:]]+[^\#]*(pam_unix|pam_pwhistory)\.so"
    Module="pam_unix.so or pam_pwhistory"
else
    Regex="^[[:space:]]*password[[:space:]]+[^\#]+"
    Regex=$Regex"[[:space:]]+[^\#]*pam_unix\.so"
    Module="pam_unix.so"
fi


# Make the opasswd File
if [ ! -e "$OpasswdFile" ]; then
    TouchLog=`/bin/touch $OpasswdFile 2>&1`
    if [ -n "$TouchLog" ]; then
        /bin/echo "FAILURE-1003: Could not create [$OpasswdFile]
 file/directory"
        exit 1003
    else
        SuccMsg="[$OpasswdFile] file/directory created\n"
    fi
fi

for FileName in $FileNames; do
    if [ ! -e "$FileName" ]; then
        FailMsg=$FailMsg"[$FileName] file/directory does not
 exist\n"
        continue;
    fi

    ExistedPamCrackLib=`/bin/egrep -i "$Regex" "$FileName" 2>/
dev/null`

    if [ -z "$ExistedPamCrackLib" ]; then
        FailMsg=$FailMsg"$FileName does not contain [$Module]
 module\n"
        continue;
    fi

    ParameterExisted=`/bin/echo "$ExistedPamCrackLib" | \
        /bin/egrep -i "\<$Parameter[[:space:]]*="`

    # Backup the file before editing
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="$TW_REMEDIATION_BACKUP_DIR$DirName"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            FailMsg="Could not create [$FullPath] file/directory"
            /bin/echo -e FAILURE-1003: $FailMsg
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        FailMsg="Could not backup [$FileName] file"
        /bin/echo -e FAILURE-1007: $FailMsg
        exit 1007
    fi

    # Issue commands to update the file
    if [ -z "$ParameterExisted" ]; then
        check=0;
    else
        check=1;
    fi

    UpdateLog=`(/bin/awk -F"#" 'BEGIN{OFS="#"}
        $1 ~ /'$Regex'/ {
            if(tolower($1) ~ /'$ParameterRegex'/){
                IGNORECASE=1;
                gsub(/'$ParameterRegex'/,"'$Parameter'='$Value'
 ",$0)
                IGNORECASE=0;
            }else{
                if( '$check' ~ /0/){
                    $1 = $1 " '$Parameter'='$Value' "
```

Post Remediation Category	*None*

Remediated Elements	`/etc/pam.d/system-auth`
	`/etc/pam.d/system-auth-ac`
	`/etc/security/opasswd`

Post Remediation Steps	No additional Post Remediation steps

Requirement 10 Track and Monitor All Access to Network Resources and Cardholder Data

Logging mechanisms and the ability to track user activities are critical in preventing, detecting, or minimizing the impact of a data compromise. The presence of logs in all environments allows thorough tracking, alerting, and analysis when something does go wrong. Determining the cause of a compromise is very difficult, if not impossible, without system activity logs.

10.2 Audit Trail Automation

Implement automated audit trails for all system components to reconstruct the following events:

10.2.0 Enable Audit

10.2.0. 1 Verify That the System Is Configured to Display to the User the Date and Time of the Last Logon or Access via ssh

Verify That the System Is Configured to Display to the User the Date and Time of the Last Logon or Access via ssh

Description	This test verifies that the system must display to the user the date and time of the last logon or access via ssh. Users need to be aware of activity that occurs regarding their account. Providing users with information regarding the date and time of their last successful login allows the user to determine if any unauthorized activity has occurred and gives them an opportunity to notify administrators. At ssh login, a user must be presented with the last successful login date and time.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*PrintLastLog[\ \t]+(\w+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) PrintLastLog Not equal "no"
Remediation	To remediate failure of this policy test,configure SSH daemon to display last login information. **Configuring SSH daemon to display last login information**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **PrintLastLog <value>**. 4. Set the line to **PrintLastLog yes** and save the file. 5. Restart the daemon using the **service sshd restart** command. For further details, please run the command **man sshd_config** to read man page.

10.2.0. 2 Verify the Kernel Auditing Is Active

Verify the Kernel Auditing Is Active

Description

This test verifies that Kernel auditing is active.
Audit events need to be captured on processes that start up prior to auditd, so that potential malicious activity cannot go undetected.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audisp/plugins.d/syslog.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*active[\ ]+=[\ ]+yes[\ ]*$/ (Flags:Multiline,Case insensitive,Comments mode)
Kernel Auditing Activation Exists

Remediation

To remediate failure of this policy test, configure audisp-syslog plug-in to send syslog audit events to syslog system

Configuring audisp-syslog plug-in to send syslog audit events to syslog system:

1. Become superuser or assume an equivalent role.
2. Run the **/bin/rpm -qa | /bin/egrep audispd** to determine whether audisp-syslog plug-in was installed
3. If it has not been installed, run **yum install audispd-plugins** to install
4. Open the **/etc/audisp/plugins.d/syslog.conf** file.
5. Find the **active** parameter and change its value to **yes** or add it if not found:
   active = yes
   Note: It requires at least a space before and after equal sign (=)
6. Save the file.
7. Run the **/sbin/service auditd restart** to apply the change.

10.2.0. 3 Verify That sshd_config Contains 'LogLevel INFO'

Verify That sshd_config Contains 'LogLevel INFO'

Description	This test verifies that the local SSH server contains 'LogLevel INFO'. The option LogLevel specifies the level that is used when logging messages from sshd.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*LogLevel[\ \t]+(\w+)[\ \t]*$/ (Flags:Multiline,Comments mode) (LogLevel Equals "INFO" AND SSH Server LogLevel Setting Exists) OR SSH Server LogLevel Setting Does not exist
Remediation	To remediate failure of this policy test, set the verbosity level that is used when logging messages from sshd to INFO. **Setting the verbosity level that is used when logging messages from sshd to INFO:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line that contains **LogLevel <value>**. 4. Set the **<value>** to **INFO** and save the file. 5. Run the **service sshd restart** command to apply the change. For further details, please run the command **man sshd_config** to read man page.

10.2.0. 4 Verify That Processes That Start Prior to auditd Are Also Audited

Verify That Processes That Start Prior to auditd Are Also Audited

Description	Audit events need to be captured on processes that start up prior to auditd, so that potential malicious activity cannot go undetected.
Severity	0
Weight	5
Type	Content Test
Rules	Content of grub.conf File
Element	Equals "Content of grub.conf File"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*kernel[\ \t]+(?!.*\baudit=1\b).*$/ (Flags:Multiline,Comments mode) Processes without Audit Does not exist
Remediation	To remediate failure of this policy test, enable auditing for processes that start prior to auditd. **Configuring auditing for processes that start prior to auditd:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/grub.conf** file. 3. Find the line that starts with **kernel**. 4. Add the **audit=1** parameter to the end of the line and save the file. For further details, please refer to: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/sec-sel-analystcontrol.html

10.2.0. 5 Verify That the rsyslog Package Is Installed

Verify That the rsyslog Package Is Installed

Description	This test verifies that rsyslog package is installed on the system. The security enhancements of rsyslog such as connection-oriented (i.e. TCP) transmission of logs, the option to log to database formats, and the encryption of log data en route to a central logging server) justify installing and configuring the package.
Severity	0
Weight	5
Type	Content Test
Rules	List of Installed Packages
Element	Equals "installed packages"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*rsyslog-\d.*$/ (Flags:Multiline,Case insensitive,Comments mode) rsyslog Installed Exists
Remediation	To remediate failure of this policy test, install rsyslog. **Install rsyslog**: 1. Become superuser or assume an equivalent role. 2. Install **rsyslog** using **yum** command: **yum install <rsyslog_pakage>** For further details, please run the command **man yum** to read man page.

10.2.0. 6 Verify That rsyslog Service Is Enabled

Verify That rsyslog Service Is Enabled

Description	This test verifies that rsyslog service is enabled. The security enhancements of rsyslog such as connection-oriented (i.e. TCP) transmission of logs, the option to log to database formats, and the encryption of log data en route to a central logging server) justify installing and configuring the package.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*rsyslog[\ \t]+.*[\ \t]+3:on[\ \t]+.*[\ \t]+5:on[\ \t]+.*$/ (Flags:Multiline,Comments mode) rsyslog Service Status Exists
Remediation	To remediate failure of this policy test, turn on the rsyslog service. **Turning on the rsyslog service:** 1. Become superuser or assume an equivalent role. 2. Run the **chkconfig --list rsyslog** command to check the service status. 3. Turn on the **rsyslog** service using the **chkconfig --level 35 rsyslog on** command. For further details, please run the command **man chkconfig** to read man page.

10.2.0. 7 Verify That the auditd Service Is Enabled

Verify That the auditd Service Is Enabled

Description	This test determines whether the auditd daemon is in a running state. This setting supports service availability and host/network integrity by ensuring that specific user/process actions are being audited.
Severity	0
Weight	5
Type	Content Test
Rules	Service Status
Element	Equals "service status"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*auditd[\ \t]+.*on.*$\n(?:^.*$\n)*?^[\ \t]*auditd[\ \t]+\(pid[\ \t]+\d+\)[\ \t]+is[\ \t]+running.*$/ (Flags:Multiline,Comments mode) auditd Service Status Exists
Remediation	To remediate failure of this policy test, turn on the auditd service. **Turning on the auditd service**: 1. Become superuser or assume an equivalent role. 2. Run the **/sbin/service auditd start** command to turn on the **auditd** service. 3. Run the **chkconfig auditd on** command to keep the **auditd** service turned on in the next reboot. For further details, please run the command **man auditd** to read man page.

10.2.0. 8 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/localtime File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/localtime File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/localtime -p wa -k time-change' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/localtime[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+time-change\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/localtime File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify system date and/or time.

Configuring the system to audit events that modify system date and/or time on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/localtime -p wa -k time-change** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify system date and/or time on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/localtime -p wa -k time-change** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/localtime -p wa -k time-change"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015496
# AR_TEST_NAME = '-w /etc/localtime -p wa -k time-change' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0. 9 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/group File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/group File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /etc/group -p wa -k identity' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/etc/group[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+identity\b).*/ (Flags:Multiline,Comments mode) audit Line for Logging Write and Attribute Change Events Relating to the /etc/group File Exists
Remediation	To remediate failure of this policy test, configure the system to audit events that modify user/group information. **Configuring the system to audit events that modify user/group information on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /etc/group -p wa -k identity** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit events that modify user/group information on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /etc/group -p wa -k identity** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the **man auditctl** command to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/group -p wa -k identity"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015498
# AR_TEST_NAME = '-w /etc/group -p wa -k identity' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.10 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/passwd File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/passwd File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/passwd -p wa -k identity' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/passwd[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+identity\b).*/
(Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/passwd File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify user/group information.

Configuring the system to audit events that modify user/group information on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/passwd -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify user/group information on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/passwd -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the **man auditctl** command to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/passwd -p wa -k identity"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015499
# AR_TEST_NAME = '-w /etc/passwd -p wa -k identity' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.11 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/gshadow File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/gshadow File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/gshadow -p wa -k identity' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/gshadow[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+identity\b).*/
(Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/gshadow File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify user/group information.

Configuring the system to audit events that modify user/group information on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/gshadow -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify user/group information on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/gshadow -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the **man auditctl** command to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/gshadow -p wa -k identity"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015500
# AR_TEST_NAME = '-w /etc/gshadow -p wa -k identity' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.12 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/shadow File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/shadow File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/shadow -p wa -k identity' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/shadow[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+identity\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/shadow File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify user/group information.

Configuring the system to audit events that modify user/group information on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/shadow -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify user/group information on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/shadow -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the **man auditctl** command to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/shadow -p wa -k identity"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015501
# AR_TEST_NAME = '-w /etc/shadow -p wa -k identity' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.13 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/security/opasswd File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/security/opasswd File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/security/opasswd -p wa -k identity' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/security/opasswd[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+identity\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/security/opasswd File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify user/group information.

Configuring the system to audit events that modify user/group information on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/security/opasswd -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify user/group information on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/security/opasswd -p wa -k identity** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/security/opasswd -p wa -k identity"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015502
# AR_TEST_NAME = '-w /etc/security/opasswd -p wa -k identity'
 Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.14 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/issue File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/issue File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/issue -p wa -k system-locale' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/issue[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+system-locale\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/issue File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify the system's network environment.

Configuring the system to audit events that modify the system's network environment on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/issue -p wa -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify the system's network environment on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/issue -p wa -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/issue -p wa -k system-locale"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015505
# AR_TEST_NAME = '-w /etc/issue -p wa -k system-locale' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.15 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/issue.net File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/issue.net File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /etc/issue.net -p wa -k system-locale' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/etc/issue\.net[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+system-locale\b).*/ (Flags:Multiline,Comments mode) audit Line for Logging Write and Attribute Change Events Relating to the /etc/issue.net File Exists
Remediation	To remediate failure of this policy test, configure the system to audit events that modify the system's network environment. **Configuring the system to audit events that modify the system's network environment on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /etc/issue.net -p wa -k system-locale** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit events that modify the system's network environment on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /etc/issue.net -p wa -k system-locale** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/issue.net -p wa -k system-locale"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015506
# AR_TEST_NAME = '-w /etc/issue.net -p wa -k system-locale'
 Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.16 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/hosts File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/hosts File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/hosts -p wa -k system-locale' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/hosts[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+system-locale\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/hosts File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify the system's network environment.

Configuring the system to audit events that modify the system's network environment on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/hosts -p wa -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify the system's network environment on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/hosts -p wa -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/hosts -p wa -k system-locale"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015507
# AR_TEST_NAME = '-w /etc/hosts -p wa -k system-locale' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.17 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/sysconfig/network File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/sysconfig/network File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/sysconfig/network -p wa -k system-locale' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/sysconfig/network[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+system-locale\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/sysconfig/network File Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify the system's network environment.

Configuring the system to audit events that modify the system's network environment on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file to add audit rules.
3. Find the line that contains the **-w /etc/sysconfig/network -p wa -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **/usr/sbin/service auditd restart** command to apply the change.

Configuring the system to audit events that modify the system's network environment on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file to add audit rules.
3. Find the line that contains the **-w /etc/sysconfig/network -p wa -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **/usr/sbin/service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/sysconfig/network -p wa -k system-locale"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015508
# AR_TEST_NAME = '-w /etc/sysconfig/network -p wa -k system-
locale' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category	`Reload Configuration "auditd"`
Remediated Elements	*None*
Post Remediation Steps	To complete this remediation: 1. Become superuser or assume an equivalent role. 2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.18 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/selinux Directory and It's Sub-directories

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/selinux Directory and It's Sub-directories

Description

This test verifies that /etc/audit/audit.rules contains the '-w /etc/selinux/ -p wa -k MAC-policy' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/etc/selinux/[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+MAC-policy\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /etc/selinux Directory and It's Sub-directories Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify the system's network environment.

Configuring the system to audit events that modify the system's network environment on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /etc/selinux/ -p wa -k MAC-policy** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify the system's network environment on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /etc/selinux/ -p wa -k MAC-policy** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-w /etc/selinux/ -p wa -k MAC-policy"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015509
# AR_TEST_NAME = '-w /etc/selinux/ -p wa -k MAC-policy' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.19 Turns on the Auditing Subsystem

Turns on the Auditing Subsystem

Description

This test verifies that auditing is enabled for this host. Make the configuration immutable - reboot is required to change audit rules.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-e[\ ]+(\d+)[\ ]*$/ (Flags:Multiline,Comments mode)
auditd Status Equals 2

Remediation

To remediate failure of this policy test, configure system to audit the loading and unloading of kernel modules.

Configuring system to audit the loading and unloading of kernel modules on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains **-e <value>**.
4. Set the **<value>** to **2** and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring system to audit the loading and unloading of kernel modules on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains **-e <value>**.
4. Set the **<value>** to **2** and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditd** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
ParameterName="-e"
SeparateSymbol=" "
Value="2"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
fi

# Issue the command to update the value of parameter
IsExisted=`/bin/awk -F"$SeparateSymbol" '$1 ~ \
    /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {print}' \
        "$FileName" 2>/dev/null`
if [ -n "$IsExisted" ]; then
    UpdateLog=`(/bin/awk -F"$SeparateSymbol" \
    '$1 ~ /^[[:space:]]*'"$ParameterName"'[[:space:]]*$/ {
        $0 = "'"$ParameterName"''"$SeparateSymbol"''"$Value"'"
    }{print}' "$BackupName" > "$FileName") 2>&1`

    # Rollback to the original file
    if [ -n "$UpdateLog" ]; then
        /bin/echo "FAILURE-4001: Could not change value of
 [$ParameterName]" \
            "parameter to [$Value] in ["$FileName"] file"
        /bin/cp -f "$BackupName" "$FileName" 2>/dev/null
        exit 4001
    fi
    /bin/echo "SUCCESS-4001: Value of [$ParameterName] parameter
 changed to" \
        "[$Value] in ["$FileName"] file"
else
    AddLog=`(/bin/echo
 "${ParameterName}${SeparateSymbol}${Value}" \
        >> "$FileName") 2>&1`
    if [ -n "$AddLog" ]; then
        /bin/echo "FAILURE-6001: Could not add"\
            "[${ParameterName}${SeparateSymbol}${Value}] line to"
 \
                "["$FileName"] file"
        exit 6001
    fi
    /bin/echo "SUCCESS-6003:
 [${ParameterName}${SeparateSymbol}${Value}]" \
        "line added to ["$FileName"] file"
fi
exit 0

# AR_ACTION = RHEL_PARAMETER_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015526
# AR_TEST_NAME = Turns on the Auditing Subsystem

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category Reload Configuration "auditd"

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.0.20 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/faillog File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/faillog File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /var/log/faillog -p wa -k logins' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/var/log/faillog[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+logins\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Write and Attribute Change Events Relating to the /var/log/faillog File Exists
Remediation	To remediate failure of this policy test, configure the system to audit the events that relate to login and logout activities. **Configuring the system to audit the events that relate to login and logout activities on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /var/log/faillog -p wa -k logins** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit the events that relate to login and logout activities on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /var/log/faillog -p wa -k logins** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.0.21 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/lastlog File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/lastlog File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /var/log/lastlog -p wa -k logins' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/var/log/lastlog[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+logins\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /var/log/lastlog File Exists

Remediation

To remediate failure of this policy test, configure the system to audit the events that relate to login and logout activities.

Configuring the system to audit the events that relate to login and logout activities on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /var/log/lastlog -p wa -k logins** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit the events that relate to login and logout activities on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /var/log/lastlog -p wa -k logins** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.0.22 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/tallylog File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/tallylog File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /var/log/tallylog -p -wa -k logins' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/var/log/tallylog[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+logins\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /var/log/tallylog File Exists

Remediation

To remediate failure of this policy test, configure the system to audit the events that relate to login and logout activities.

Configuring the system to audit the events that relate to login and logout activities on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /var/log/tallylog -p wa -k logins** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit the events that relate to login and logout activities on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /var/log/tallylog -p wa -k logins** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.0.23 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/btmp File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/btmp File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /var/log/btmp -p wa -k session' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/var/log/btmp[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+session\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Write and Attribute Change Events Relating to the /var/log/btmp File Exists
Remediation	To remediate failure of this policy test, configure the system to audit session initiation events. **Configuring the system to audit session initiation events on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /var/log/btmp -p wa -k session** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit session initiation events on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /var/log/btmp -p wa -k session** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.0.24 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/run/utmp File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/run/utmp File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /var/run/utmp -p wa -k session' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/var/run/utmp[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+session\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /var/run/utmp File Exists

Remediation

To remediate failure of this policy test, configure the system to audit session initiation events.

Configuring the system to audit session initiation events on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /var/run/utmp -p wa -k session** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit session initiation events on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /var/run/utmp -p wa -k session** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.0.25 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/wtmp File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/wtmp File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /var/log/wtmp -p wa -k session' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/var/log/wtmp[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+session\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /var/log/wtmp File Exists

Remediation

To remediate failure of this policy test, configure the system to audit session initiation events.

Configuring the system to audit session initiation events on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /var/log/wtmp -p wa -k session** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

To remediate failure of this policy test, configure system to audit session initiation events.

Configuring the system to audit session initiation events on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /var/log/wtmp -p wa -k session** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.0.26 For 64 Bit Architecture: Verify That audit Logging Is Enabled on the mount Events by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled on the mount Events by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S mount -F auid>=500 -F auid!=4294967295 -k mounts' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+mount\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+mounts\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging the mount Events by Users Exists

Remediation

To remediate failure of this policy test, configure system to audit successful file system mounts.

Configuring system to audit successful file system mounts on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S mount -F auid>=500 -F auid!=4294967295 -k mounts** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring system to audit successful file system mounts on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S mount -F auid>=500 -F auid!=4294967295 -k mounts** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

10.2.0.27 For 32 Bit Architecture: Verify That audit Logging Is Enabled on the mount Events by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled on the mount Events by Users

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 CentOS Linux release 7.2.1511
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+mount\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+mounts\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging the mount Events by Users Exists
Remediation	To remediate failure of this policy test, configure system to audit successful file system mounts. **Configuring system to audit successful file system mounts on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring system to audit successful file system mounts on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.0.28 Verify That rsyslog Is Configured to Send Logs to a Remote Log Host

Verify That rsyslog Is Configured to Send Logs to a Remote Log Host

Description

This test verifies that rsyslogd is configured to send logs to a remote loghost.
Storing log data on a remote host protects log integrity from local attacks. If an attacker gains root access on the local system, they could tamper with or remove log data that is stored on the local system

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Element

Equals "/etc/rsyslog.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]**\.*[\ \t]+(?:@|:omrelp:)\S+[\ \t]*(?:$|\#)/ (Flags:Multiline,Case insensitive,Comments mode)
Send Logs to a Remote Log Host Setting Exists

Remediation

To remediate failure of this policy test, configure the /etc/rsyslog.conf file to send logs to a remote log host.

Configuring the /etc/rsyslog.conf file to send logs to a remote log host:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/rsyslog.conf** file.
3. Review the file:
   - Add the following to the file if the system uses UDP for log message delivery:
     ***.* @[loghost.example.com]**
   - Add the following to the file if the system uses TCP for log message delivery:
     ***.* @@[loghost.example.com]**
   - Add the following to the file if the system uses RELP for log message delivery:
     ***.* :omrelp:[loghost.example.com]**

   where **[loghost.example.com]** is a remote log host.
4. Run the **service rsyslog restart** command to apply changes.

For further details, please refer to:

http://www.rsyslog.com/doc/rsyslog_conf.html

10.2.0.29 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/sudo.log File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /var/log/sudo.log File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /var/log/sudo.log -p wa -k actions' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/var/log/sudo\.log[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+actions\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging Write and Attribute Change Events Relating to the /var/log/sudo.log File Exists

Remediation

To remediate failure of this policy test, configure the system to audit system administrator actions.

Configuring the system to audit system administrator actions on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /var/log/sudo.log -p wa -k actions** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit system administrator actions on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /var/log/sudo.log -p wa -k actions** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.0.30 Verify That audit Logging Is Enabled to Log Execute Events Relating to the /sbin/insmod File

Verify That audit Logging Is Enabled to Log Execute Events Relating to the /sbin/insmod File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /sbin/insmod -p x -k modules' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/sbin/insmod[\ ]+(?=.*-p[\ ]+x\b)(?=.*-k[\ ]+modules\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Execute Events Relating to the /sbin/insmod File Exists
Remediation	To remediate failure of this policy test, configure system to audit the loading and unload ing of kernel modules. **Configuring system to audit the loading and unloading of kernel modules on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /sbin/insmod -p x -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring system to audit the loading and unloading of kernel modules on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /sbin/insmod -p x -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.0.31 Verify That audit Logging Is Enabled to Log Execute Events Relating to the /sbin/rmmod File

Verify That audit Logging Is Enabled to Log Execute Events Relating to the /sbin/rmmod File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /sbin/rmmod -p x -k modules' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/sbin/rmmod[\ ]+(?=.*-p[\ ]+x\b)(?=.*-k[\ ]+modules\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Execute Events Relating to the /sbin/rmmod File Exists
Remediation	To remediate failure of this policy test, configure system to audit the loading and unloading of kernel modules. **Configuring system to audit the loading and unloading of kernel modules on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /sbin/rmmod -p x -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring system to audit the loading and unloading of kernel modules on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /sbin/rmmod -p x -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.0.32 Verify That audit Logging Is Enabled to Log Execute Events Relating to the /sbin/modprobe File

Verify That audit Logging Is Enabled to Log Execute Events Relating to the /sbin/modprobe File

Description

This test verifies that /etc/audit/audit.rules contains the '-w /sbin/modprobe -p x -k modules' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-w[\ ]+/sbin/modprobe[\ ]+(?=.*-p[\ ]+x\b)(?=.*-k[\ ]+modules\b).*$/ (Flags:Multiline,Comments mode)
audit Line for Logging Execute Events Relating to the /sbin/modprobe File Exists

Remediation

To remediate failure of this policy test, configure system to audit the loading and unloading of kernel modules.

Configuring system to audit the loading and unloading of kernel modules on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-w /sbin/modprobe -p x -k modules** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring system to audit the loading and unloading of kernel modules on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-w /sbin/modprobe -p x -k modules** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.0.33 Verify That an Audit Line for Each setuid/setgid Program Appears in the Audit File

Verify That an Audit Line for Each setuid/setgid Program Appears in the Audit File

Description

Execution of privileged commands by non-privileged users could be an indication of someone trying to gain unauthorized access to the system.
Monitor privileged programs (those that have the setuid and/or setgid bit set on execu tion) to determine if unprivileged users are running these commands.

Severity

0

Weight

5

Type

Content Test

Rules

Verify That an Audit Line for Each setuid/setgid Program Identified

Excluded Nodes

Red Hat Enterprise Linux Server 6

CentOS 6

Element

Equals "Audit Line for setuid/setgid Programs"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
Privileged Programs without Audit Line Does not exist

Remediation

To remediate failure of this policy test, configure each setuid/setgid program has an audit line in the audit file.

Adding an audit line for each setuid/setgid program appears in the audit file:

1. Become superuser or assume an equivalent role.
2. Run the following script to list all the setuid/setgid programs which have not audit line in the audit file:

```
FileNames=`/usr/bin/find / -xdev \( -perm -4000 -o -perm -2000
\) -type f 2>/dev/null`;if [ -n "$FileNames" ]; then for FileName
in $FileNames; do Regex="`/bin/echo $FileName | /bin/sed
's/[\.\/]/\\\&/g'`"; IsExisted=`/sbin/auditctl -l 2>/dev/null |/bin/
awk '$0 ~ /^[[:space:]]*LIST_RULES[[:space:]]*:[[:space:]]*exi
t,always/ && $0 ~ /[[:space:]]watch='"$Regex"'[[:space:]]+/
&& $0 ~ /[[:space:]]perm=[[:graph:]]*x[[:graph:]]*[[:space:]]+/
&& $0 ~ /[[:space:]]auid>=500[[:space:]]+/ && $0 ~ /[[:space:]
]auid!=-1[[:space:]]+/ {print $0}'`; if [ -z "$IsExisted" ]; then /
bin/echo "$FileName"; fi; done;fi
```

3. Open the **/etc/audit/audit.rules** file.
4. For each **<FileName>** listed in the step 2, add the following entry to the end of file:

```
-a always,exit -F path=<FileName> -F perm=x -F auid>=500 -F
auid!=4294967295 -k privileged
```

5. Save the file.
6. Run the **/sbin/service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

10.2.2 Privileged User Action

All actions taken by any individual with root or administrative privileges.

10.2.2. 1 For 32 Bit Architecture: Verify That audit Logging Is Enabled to Log Events to Tune Kernel Clock, Set Time

For 32 Bit Architecture: Verify That audit Logging Is Enabled to Log Events to Tune Kernel Clock, Set Time

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 CentOS Linux release 7.2.1511
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+adjtimex\b)(?=.*-S[\ ]+settimeofday\b)(?=.*-S[\ ]+stime\b)(?=.*-k[\ ]+time-change\b).*/ (Flags:Multiline,Comments mode) audit Line for Logging Events to Tune Kernel Clock, Set Time Exists
Remediation	To remediate failure of this policy test, configure the system to audit events that modify system date and/or time. **Configuring the system to audit events that modify system date and/or time on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change** entry. 4. Uncomment that line or add it to the end of file (if not found) and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit events that modify system date and/or time on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change** entry. 4. Uncomment that line or add it to the end of file (if not found) and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S adjtimex -S settimeofday -S
 stime -k"
Line=$Line" time-change"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015494
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S adjtimex -S
 settimeofday -S stime -k time-change' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

Reload Configuration "auditd"

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2. 2 Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/sudoers File

Verify That audit Logging Is Enabled to Log Write and Attribute Change Events Relating to the /etc/sudoers File

Description	This test verifies that /etc/audit/audit.rules contains the '-w /etc/sudoers -p wa -k scope' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-w[\ ]+/etc/sudoers[\ ]+(?=.*-p[\ ]+wa\b)(?=.*-k[\ ]+scope\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Write and Attribute Change Events Relating to the /etc/sudoers File Exists
Remediation	To remediate failure of this policy test, configure the system to audit events that changes to system administration scope. **Configuring the system to audit events that changes to system administration scope on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-w /etc/sudoers -p wa -k scope** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit events that changes to system administration scope on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-w /etc/sudoers -p wa -k scope** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.2. 3 For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Events to Initialize or Delete Modules

For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Events to Initialize or Delete Modules

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S init_module -S delete_module -k modules' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	Audit Rules for 64 bits Architecture
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+init_module\b)(?=.*-S[\ ]+delete_module\b)(?=.*-k[\ ]+modules\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging the Events to Initialize or Delete Modules Exists
Remediation	To remediate failure of this policy test, configure system to audit the loading and unloading of kernel modules. **Configuring system to audit the loading and unloading of kernel modules on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b64 -S init_module -S delete_module -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring system to audit the loading and unloading of kernel modules on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b64 -S init_module -S delete_module -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Note**: This configuration only applies to 64 bits architecture. For further details, please run the command **man auditctl**to read man page.

10.2.2. 4 For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Events to Initialize or Delete Modules

For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Events to Initialize or Delete Modules

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S init_module -S delete_module -k modules' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+init_module\b)(?=.*-S[\ ]+delete_module\b)(?=.*-k[\ ]+modules\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging the Events to Initialize or Delete Modules Exists
Remediation	To remediate failure of this policy test, configure system to audit the loading and unloading of kernel modules. **Configuring system to audit the loading and unloading of kernel modules on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S init_module -S delete_module -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring system to audit the loading and unloading of kernel modules on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S init_module -S delete_module -k modules** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.2. 5 For 64 Bit Architecture: Verify That audit Logging Is Enabled to Log Events of clock_settime() Functions

For 64 Bit Architecture: Verify That audit Logging Is Enabled to Log Events of clock_settime() Functions

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S clock_settime -k time-change' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	Audit Rules for 64 bits Architecture
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 CentOS Linux release 7.2.1511 CentOS 5
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+clock_settime\b)(?=.*-k[\ ]+time-change\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Events of clock_settime() Functions Exists
Remediation	To remediate failure of this policy test, configure the system to audit events that modify system date and/or time. **Configuring the system to audit events that modify system date and/or time on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b64 -S clock_settime -k time-change** entry. 4. Uncomment that line or add it to the end of file (if not found) then save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit events that modify system date and/or time on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b64 -S clock_settime -k time-change** entry. 4. Uncomment that line or add it to the end of file (if not found) then save the file. 5. Run the **service auditd restart** command to apply the change. **Note**: This configuration only applies to 64 bits architecture. For further details, please run the command **man auditctl** to read man page.

10.2.2. 6 For 32 Bit Architecture: Verify That audit Logging Is Enabled to Log Events of clock_settime() Functions

For 32 Bit Architecture: Verify That audit Logging Is Enabled to Log Events of clock_settime() Functions

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S clock_settime -k time-change' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 CentOS Linux release 7.2.1511 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+clock_settime\b)(?=.*-k[\ ]+time-change\b).*$/ (Flags:Multiline,Comments mode) audit Line for Logging Events of clock_settime() Functions Exists
Remediation	To remediate failure of this policy test, configure the system to audit events that modify system date and/or time. **Configuring the system to audit events that modify system date and/or time on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S clock_settime -k time-change** entry. 4. Uncomment that line or add it to the end of file (if not found) then save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit events that modify system date and/or time on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S clock_settime -k time-change** entry. 4. Uncomment that line or add it to the end of file (if not found) then save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.

10.2.2. 7 For 64 Bit Architecture: Verify That audit Logging Is Enabled to Log Events to Tune Kernel Clock, Set Time

For 64 Bit Architecture: Verify That audit Logging Is Enabled to Log Events to Tune Kernel Clock, Set Time

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S adjtimex -S settimeofday -k time-change' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

Red Hat Enterprise Linux Server 6

CentOS Linux release 7.2.1511

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+adjtimex\b)(?=.*-S[\ ]+settimeofday\b)(?=.*-k[\ ]+time-change\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Events to Tune Kernel Clock, Set Time Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify system date and/or time.

Configuring system to audit events that modify system date and/or time on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S adjtimex -S set timeofday -k time-change** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify system date and/or time on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S adjtimex -S set timeofday -k time-change** entry.
4. Uncomment that line or add it to the end of file (if not found) and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S adjtimex -S settimeofday -k
 time-change"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015495
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S adjtimex -S
 settimeofday -k time-change' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category	`Reload Configuration "auditd"`
Remediated Elements	`/etc/audit/audit.rules`
Post Remediation Steps	To complete this remediation: 1. Become superuser or assume an equivalent role. 2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2. 8 For 32 Bit Architecture: Verify That audit Logging Is Enabled for Host Name and Domain Name Settings

For 32 Bit Architecture: Verify That audit Logging Is Enabled for Host Name and Domain Name Settings

Description

This test verifies that /etc/audit/audit.rules contains the '-a exit,always -F arch=b32 -S sethostname -S setdomainname -k system-locale' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

Red Hat Enterprise Linux Server 5

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+sethostname\b)(?=.*-S[\ ]+setdomainname\b)(?=.*-k[\ ]+system-locale\b).*/ (Flags:Multiline,Comments mode)
audit Line for Host Name and Domain Name Settings Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify the system's network environment.

Configuring the system to audit events that modify the system's network environment on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a exit,always -F arch=b32 -S sethostname -S setdomainname -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify the system's network environment on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a exit,always -F arch=b32 -S sethostname -S setdomainname -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a exit,always -F arch=b32 -S sethostname -S setdomainname
 -k syst"
Line=$Line"em-locale"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015503
# AR_TEST_NAME = '-a exit,always -F arch=b32 -S sethostname -S
 setdomainname -k system-locale' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

```
Reload Configuration "auditd"
```

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2. 9 For 64 Bit Architecture: Verify That audit Logging Is Enabled for Host Name and Domain Name Settings

For 64 Bit Architecture: Verify That audit Logging Is Enabled for Host Name and Domain Name Settings

Description

This test verifies that /etc/audit/audit.rules contains the '-a exit,always -F arch=b64 -S sethostname -S setdomainname -k system-locale' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+sethostname\b)(?=.*-S[\ ]+setdomainname\b)(?=.*-k[\ ]+system-locale\b).*/ (Flags:Multiline,Comments mode)
audit Line for Host Name and Domain Name Settings Exists

Remediation

To remediate failure of this policy test, configure the system to audit events that modify the system's network environment.

Configuring the system to audit events that modify the system's network environment on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a exit,always -F arch=b64 -S sethostname -S setdomainname -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit events that modify the system's network environment on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a exit,always -F arch=b64 -S sethostname -S setdomainname -k system-locale** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a exit,always -F arch=b64 -S sethostname -S setdomainname
 -k syst"
Line=$Line"em-locale"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015504
# AR_TEST_NAME = '-a exit,always -F arch=b64 -S sethostname -S
 setdomainname -k system-locale' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category Reload Configuration "auditd"

Remediated Elements /etc/audit/audit.rules

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.10 For 32 Bit Architecture: Verify That audit Logging Is Enabled for Permission Changes by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled for Permission Changes by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+chmod\b)(?=.*-S[\ ]+fchmod\b)(?=.*-S[\ ]+fchmodat\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+perm_mod\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Permission Changes by Users Exists

Remediation

To remediate failure of this policy test, configure the system to audit the events that modify access control permission.

Configuring the system to audit the events that modify access control permission on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit the events that modify access control permission on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -
F auid>"
Line=$Line"=500 -F auid!=4294967295 -k perm_mod"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015510
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S chmod -S fchmod -
S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.11 For 64 Bit Architecture: Verify That audit Logging Is Enabled for Permission Changes by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled for Permission Changes by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S chmod -S fchmod -S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+chmod\b)(?=.*-S[\ ]+fchmod\b)(?=.*-S[\ ]+fchmodat\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+perm_mod\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Permission Changes by Users Exists

Remediation

To remediate failure of this policy test, configure the system to audit the events that modify access control permission.

Configuring the system to audit the events that modify access control permission on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S chmod -S fchmod -S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit the events that modify access control permission on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the**/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S chmod -S fchmod -S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S chmod -S fchmod -S fchmodat -
F auid>"
Line=$Line"=500 -F auid!=4294967295 -k perm_mod"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015511
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S chmod -S fchmod -
S fchmodat -F auid>=500 -F auid!=4294967295 -k perm_mod' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements `/etc/audit/audit.rules`

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.12 For 32 Bit Architecture: Verify That audit Logging Is Enabled for Owner Changes by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled for Owner Changes by Users

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k perm_mod' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 CentOS Linux release 7.2.1511
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+chown\b)(?=.*-S[\ ]+fchown\b)(?=.*-S[\ ]+fchownat\b)(?=.*-S[\ ]+lchown\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+perm_mod\b).*/ (Flags:Multiline,Comments mode) audit Line for Logging Owner Changes by Users Exists
Remediation	To remediate failure of this policy test, configure the system to audit the events that modify access control permission. **Configuring the system to audit the events that modify access control permission on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k perm_mod** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit the events that modify access control permission on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k perm_mod** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S chown -S fchown -S fchownat -
S lchow"
Line=$Line"n -F auid>=500 -F auid!=4294967295 -k perm_mod"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015512
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S chown -S fchown
 -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k
 perm_mod' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.13 For 64 Bit Architecture: Verify That audit Logging Is Enabled for Owner Changes by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled for Owner Changes by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S chown -S fchown -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k perm_mod' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+chown\b)(?=.*-S[\ ]+fchown\b)(?=.*-S[\ ]+fchownat\b)(?=.*-S[\ ]+lchown\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+perm_mod\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Owner Changes by Users Exists

Remediation

To remediate failure of this policy test, configure the system to audit the events that modify access control permission.

Configuring the system to audit the events that modify access control permission on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S chown -S fchown -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit the events that modify access control permission on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the**/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S chown -S fchown -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S chown -S fchown -S fchownat -
S lchow"
Line=$Line"n -F auid>=500 -F auid!=4294967295 -k perm_mod"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015513
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S chown -S fchown
 -S fchownat -S lchown -F auid>=500 -F auid!=4294967295 -k
 perm_mod' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements `/etc/audit/audit.rules`

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.14 For 32 Bit Architecture: Verify That audit Logging Is Enabled for Changes in Extended File Attributes by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled for Changes in Extended File Attributes by Users

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 CentOS Linux release 7.2.1511
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+setxattr\b)(?=.*-S[\ ]+lsetxattr\b)(?=.*-S[\ ]+fsetxattr\b)(?=.*-S[\ ]+removexattr\b)(?=.*-S[\ ]+lremovexattr\b)(?=.*-S[\ ]+fremovexattr\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+perm_mod\b).*/ (Flags:Multiline,Comments mode) audit Line for Logging Changes in Extended File Attributes by Users Exists
Remediation	To remediate failure of this policy test, configure the system to audit the events that modify access control permission. **Configuring the system to audit the events that modify access control permission on RHEL 5, 6**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit the events that modify access control permission on RHEL 7**: 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S setxattr -S lsetxattr -S
 fsetxattr -"
Line=$Line"S removexattr -S lremovexattr -S fremovexattr -F
 auid>=500 -F auid"
Line=$Line"!=4294967295 -k perm_mod"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015514
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S setxattr -
S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S
 fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod'
 Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements *None*

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.15 For 64 Bit Architecture: Verify That audit Logging Is Enabled for Changes in Extended File Attributes by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled for Changes in Extended File Attributes by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+setxattr\b)(?=.*-S[\ ]+lsetxattr\b)(?=.*-S[\ ]+fsetxattr\b)(?=.*-S[\ ]+removexattr\b)(?=.*-S[\ ]+lremovexattr\b)(?=.*-S[\ ]+fremovexattr\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+perm_mod\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging Changes in Extended File Attributes by Users Exists

Remediation

To remediate failure of this policy test, configure the system to audit the events that modify access control permission.

Configuring the system to audit the events that modify access control permission on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit the events that modify access control permission on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S setxattr -S lsetxattr -S
 fsetxattr -"
Line=$Line"S removexattr -S lremovexattr -S fremovexattr -F
 auid>=500 -F auid"
Line=$Line"!=4294967295 -k perm_mod"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015515
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S setxattr -
S lsetxattr -S fsetxattr -S removexattr -S lremovexattr -S
 fremovexattr -F auid>=500 -F auid!=4294967295 -k perm_mod'
 Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements `/etc/audit/audit.rules`

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.16 For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Access Deny Failures to Create, Open or Truncate Files by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Access Deny Failures to Create, Open or Truncate Files by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+creat\b)(?=.*-S[\ ]+open\b)(?=.*-S[\ ]+openat\b)(?=.*-S[\ ]+truncate\b)(?=.*-S[\ ]+ftruncate\b)(?=.*-F[\ ]+exit[\ ]*=[\ ]*-EACCES\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+access\b).*/ (Flags:Multiline,Comments mode)
The Audit System Logs Failed Access Attempts of Normal Users Using Create, Open or Truncate Command to Files and Programs Exists

Remediation

To remediate failure of this policy test, configure the system to audit unsuccessful unauthorized access attempts to files.

Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S creat -S open -S openat -S
 truncate "
Line=$Line"-S ftruncate -F exit=-EACCES -F auid>=500 -F auid!
=4294967295 -k a"
Line=$Line"ccess"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015516
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S creat -S open -S
 openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F
 auid!=4294967295 -k access' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.17 For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Insufficient Privilege Failures to Create, Open or Truncate Files by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Insufficient Privilege Failures to Create, Open or Truncate Files by Users

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 CentOS Linux release 7.2.1511
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+creat\b)(?=.*-S[\ ]+open\b)(?=.*-S[\ ]+openat\b)(?=.*-S[\ ]+truncate\b)(?=.*-S[\ ]+ftruncate\b)(?=.*-F[\ ]+exit[\ ]*=[\ ]*-EPERM\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+access\b).*/ (Flags:Multiline,Comments mode) The Audit System Logs Failed Operation Attempts of Normal Users Using Create, Open or Truncate Command to Files and Programs Exists
Remediation	To remediate failure of this policy test, configure the system to audit unsuccessful unauthorized access attempts to files. **Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. For further details, please run the command **man auditctl** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S creat -S open -S openat -S
 truncate "
Line=$Line"-S ftruncate -F exit=-EPERM -F auid>=500 -F auid!
=4294967295 -k ac"
Line=$Line"cess"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015517
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S creat -S open -S
 openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F
 auid!=4294967295 -k access' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.18 For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Access Deny Failures to Create, Open or Truncate Files by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Access Deny Failures to Create, Open or Truncate Files by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+creat\b)(?=.*-S[\ ]+open\b)(?=.*-S[\ ]+openat\b)(?=.*-S[\ ]+truncate\b)(?=.*-S[\ ]+ftruncate\b)(?=.*-F[\ ]+exit[\ ]*=[\ ]*-EACCES\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+access\b).*/ (Flags:Multiline,Comments mode)
The Audit System Logs Failed Access Attempts of Normal Users Using Create, Open or Truncate Command to Files and Programs Exists

Remediation

To remediate failure of this policy test, configure the system to audit unsuccessful unauthorized access attempts to files.

Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S creat -S open -S openat -S
 truncate "
Line=$Line"-S ftruncate -F exit=-EACCES -F auid>=500 -F auid!
=4294967295 -k a"
Line=$Line"ccess"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015518
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S creat -S open -S
 openat -S truncate -S ftruncate -F exit=-EACCES -F auid>=500 -F
 auid!=4294967295 -k access' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category `Reload Configuration "auditd"`

Remediated Elements `/etc/audit/audit.rules`

Post Remediation Steps To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.19 For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Insufficient Privilege Failures to Create, Open or Truncate Files by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Insufficient Privilege Failures to Create, Open or Truncate Files by Users

Description	This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access' option. It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong. This configuration only applies to 64 bits architecture.
Severity	0
Weight	5
Type	Content Test
Rules	Audit Rules for 64 bits Architecture
Excluded Nodes	CentOS Linux release 7.0.1406 Red Hat Enterprise Linux Server 7 CentOS Linux release 7.2.1511
Element	Equals "/etc/audit/audit.rules"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+creat\b)(?=.*-S[\ ]+open\b)(?=.*-S[\ ]+openat\b)(?=.*-S[\ ]+truncate\b)(?=.*-S[\ ]+ftruncate\b)(?=.*-F[\ ]+exit[\ ]*=[\ ]*-EPERM\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+access\b).*/ (Flags:Multiline,Comments mode) The Audit System Logs Failed Operation Attempts of Normal Users Using Create, Open or Truncate Command to Files and Programs Exists
Remediation	To remediate failure of this policy test, configure the system to audit unsuccessful unauthorized access attempts to files. **Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 5, 6:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Configuring the system to audit unsuccessful unauthorized access attempts to files on RHEL 7:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/rules.d/audit.rules** file. 3. Find the line that contains the **-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access** entry. 4. Uncomment that line or add if not found and save the file. 5. Run the **service auditd restart** command to apply the change. **Note**: This configuration only applies to 64 bits architecture. For further details, please run the command **man auditctl** to read man page.
Command Line	`/bin/sh $(ScriptFile.sh)`

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S creat -S open -S openat -S
 truncate "
Line=$Line"-S ftruncate -F exit=-EPERM -F auid>=500 -F auid!
=4294967295 -k ac"
Line=$Line"cess"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015519
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S creat -S open -S
 openat -S truncate -S ftruncate -F exit=-EPERM -F auid>=500 -F
 auid!=4294967295 -k access' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

`/etc/audit/audit.rules`

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.20 For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Events That Unlink or Rename Files by Users

For 32 Bit Architecture: Verify That audit Logging Is Enabled on the Events That Unlink or Rename Files by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295 -k delete' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b32\b)(?=.*-S[\ ]+unlink\b)(?=.*-S[\ ]+unlinkat\b)(?=.*-S[\ ]+rename\b)(?=.*-S[\ ]+renameat\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+delete\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging the Events That Unlink or Rename Files by Users Exists

Remediation

To remediate failure of this policy test, configure system to audit file deletion events.

Configuring system to audit file deletion events on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295 -k delete** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring system to audit file deletion events on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295 -k delete** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b32 -S unlink -S unlinkat -S rename
 -S rena"
Line=$Line"meat -F auid>=500 -F auid!=4294967295 -k delete"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015523
# AR_TEST_NAME = '-a always,exit -F arch=b32 -S unlink -S
 unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295
 -k delete' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

None

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.2.2.21 For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Events That Unlink or Rename Files by Users

For 64 Bit Architecture: Verify That audit Logging Is Enabled on the Events That Unlink or Rename Files by Users

Description

This test verifies that /etc/audit/audit.rules contains the '-a always,exit -F arch=b64 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295 -k delete' option.
It is important to maintain an audit trail in order to thoroughly track and analyze system activity when something goes wrong.
This configuration only applies to 64 bits architecture.

Severity

0

Weight

5

Type

Content Test

Rules

Audit Rules for 64 bits Architecture

Excluded Nodes

CentOS Linux release 7.0.1406

Red Hat Enterprise Linux Server 7

CentOS Linux release 7.2.1511

Element

Equals "/etc/audit/audit.rules"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^[\ ]*-a[\ ]+(?=\S*\bexit\b)(?=\S*\balways\b)(?!\S*\bentry\b)(?!\S*\bnever\b)\S*[\ ]+(?=.*-F[\ ]+arch=b64\b)(?=.*-S[\ ]+unlink\b)(?=.*-S[\ ]+unlinkat\b)(?=.*-S[\ ]+rename\b)(?=.*-S[\ ]+renameat\b)(?=.*-F[\ ]+auid>=500\b)(?=.*-F[\ ]+auid!=4294967295\b)(?=.*-k[\ ]+delete\b).*/ (Flags:Multiline,Comments mode)
audit Line for Logging the Events That Unlink or Rename Files by Users Exists

Remediation

To remediate failure of this policy test, configure system to audit file deletion events.

Configuring system to audit file deletion events on RHEL 5, 6:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295 -k delete** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Configuring system to audit file deletion events on RHEL 7:

1. Become superuser or assume an equivalent role.
2. Open the **/etc/audit/rules.d/audit.rules** file.
3. Find the line that contains the **-a always,exit -F arch=b64 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295 -k delete** entry.
4. Uncomment that line or add if not found and save the file.
5. Run the **service auditd restart** command to apply the change.

Note: This configuration only applies to 64 bits architecture.

For further details, please run the command **man auditctl** to read man page.

Command Line

```
/bin/sh $(ScriptFile.sh)
```

Script

```
# /bin/sh $(ScriptFile.sh)

# Initialize Variables
FileName="/etc/audit/audit.rules"
Line="-a always,exit -F arch=b64 -S unlink -S unlinkat -S rename
 -S rena"
Line=$Line"meat -F auid>=500 -F auid!=4294967295 -k delete"

# Backup the file before updating
if [ -e "$FileName" ]; then
    BaseName=`/bin/basename "$FileName" 2>/dev/null`
    DirName=`/usr/bin/dirname "$FileName" 2>/dev/null`
    FullPath="${TW_REMEDIATION_BACKUP_DIR}${DirName}"
    if [ ! -d "$FullPath" ]; then
        CreateLog=`/bin/mkdir -p "$FullPath" 2>&1`
        if [ -n "$CreateLog" ]; then
            /bin/echo "FAILURE-1003: Could not create"\
               "[$FullPath] file/directory"
            exit 1003
        fi
    fi
    BackupName="$FullPath/${BaseName}.tecopy"
    CopyLog=`/bin/cp -f "$FileName" "$BackupName" 2>&1`
    if [ -n "$CopyLog" ]; then
        /bin/echo "FAILURE-1007: Could not backup [$FileName]
 file"
        exit 1007
    fi
else
    /bin/echo FAILURE-1002: [$FileName] file/directory does not
 exist
    exit 1002
fi

# Issue the command to add line to the file
AddLog=`(/bin/echo "$Line" >> $FileName) 2>&1`
if [ -n "$AddLog" ]; then
    /bin/echo "FAILURE-6001: Could not add [$Line] line to
 [$FileName] file"
    exit 6001
fi
/bin/echo "SUCCESS-6003: [$Line] line added to [$FileName] file"
exit 0

# AR_ACTION = RHEL_LINE_SETTING
# AR_COMPLETION = COMPLETION_RELOAD_SERVICE auditd
# AR_TEST_ID = T0015524
# AR_TEST_NAME = '-a always,exit -F arch=b64 -S unlink -S
 unlinkat -S rename -S renameat -F auid>=500 -F auid!=4294967295
 -k delete' Option

# AR_FINAL_STEPS = To complete this remediation:
# AR_FINAL_STEPS = <ol><li>Become superuser or assume an
 equivalent role.</li><li>Run the <b>/etc/init.d/auditd reload</
b> command to reload the filters.</li>
# AR_FINAL_STEPS = </ol>
```

Post Remediation Category

`Reload Configuration "auditd"`

Remediated Elements

`/etc/audit/audit.rules`

Post Remediation Steps

To complete this remediation:

1. Become superuser or assume an equivalent role.
2. Run the **/etc/init.d/auditd reload** command to reload the filters.

10.4 Time Synchronization

Using time synchronization technology, synchronize all critical system clocks and times and ensure that the following is implemented for acquiring, distributing, and storing time.
Note: One example of time synchronization technology is Network Time Protocol (NTP).

10.4.1 Correct System Time

Critical systems have the correct and consistent time.

10.4.1.1 Verify That the System Is Configured to Use an NTP Server

Verify That the System Is Configured to Use an NTP Server

Description	This test verifies that the system clock is synchronized to a trusted time source. Synchronizing with an NTP server makes it possible to collate system logs from multiple sources or correlate computer events with real time events. Using a trusted NTP server provided by your organization is recommended.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ntp.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*server[\ \t]+\S+$/ (Flags:Multiline,Comments mode) server Exists
Remediation	To remediate failure of this policy test, config the server for the NTP to synchronize system clock: **Config the server for the NTP to synchronize system clock:** 1. Become super user or equivalent roles 2. Open **/etc/ntp.conf** file 3. Add the following line: **server <ntp-server>** <ntp-server> 4. Save and close the file

10.4.3 Trusted Time Sources

Time settings are received from industry-accepted time sources.

10.4.3.1 Verify That "restrict -6 default" Is Configured with Correct Parameters

Verify That "restrict -6 default" Is Configured with Correct Parameters

Description	The Network Time Protocol (NTP) is designed to synchronize system clocks across a variety of systems and use a source that is highly accurate. It is recommended that physical systems and virtual guests lacking direct access to the physical host's clock be configured as NTP clients to synchronize their clocks (especially to support time sensitive security mechanisms like Kerberos). This also ensures log files have consistent time records across the enterprise, which aids in forensic investigations.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Element	Equals "/etc/ntp.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*restrict[\ \t]+-6\b[\ \t]+default\b(?=.*[\ \t]kod\b)(?=.*[\ \t]nomodify\b)(?=.*[\ \t]notrap\b)(?=.*[\ \t]nopeer\b)(?=.*[\ \t]noquery\b).*$/ (Flags:Multiline,Comments mode) restrict -6 default Exists
Remediation	To remediate the failure of this policy test, set correct parameters to restrict -6 default to prevent clients from accessing to the physical host's clock **Set correct parameters to restrict -6 default** 1. Become a superuser or assume an equivalent role 2. Open **/etc/ntp.conf** file 3. Find the line that contains **restrict -6 default** entry 4. Uncomment and change it to **restrict -6 default kod nomodify nopeer notrap noquery** or add if not found 5. Save and close the file For more information, please refer to: https://support.ntp.org/bin/view/Support/AccessRestrictions

10.4.3.2 Verify That "restrict default" Is Configured with Correct Parameters

Verify That "restrict default" Is Configured with Correct Parameters

Description

This test verifies that "restrict default" is configured to "kod nomodify notrap nopeer noquery". The Network Time Protocol (NTP) is designed to synchronize system clocks across a variety of systems and use a source that is highly accurate. It is recommended that physical systems and virtual guests lacking direct access to the physical host's clock be configured as NTP clients to synchronize their clocks (especially to support time sensitive security mechanisms like Kerberos). This also ensures log files have consistent time records across the enterprise, which aids in forensic investigations.

Severity

0

Weight

5

Type

Content Test

Rules

System Configuration Files

Element

Equals "/etc/ntp.conf"

Version conditions

If an element version has no content, the condition should:Fail
Regular expression: /^[\ \t]*restrict[\ \t]+default\b(?=.*[\ \t]kod\b)(?=.*[\ \t]nomodify\b)(?=.*[\ \t]notrap\b)(?=.*[\ \t]nopeer\b)(?=.*[\ \t]noquery\b).*$/ (Flags:Multiline,Comments mode)
restrict default Exists

Remediation

To remediate the failure of this policy test, set correct parameters to restrict default to prevent clients from accessing to the physical host's clock.

Set correct parameters to restrict default:

1. Become a superuser or assume an equivalent role.
2. Open **/etc/ntp.conf** file.
3. Find the line that contains **restrict default** entry.
4. Uncomment and change it to **restrict default kod nomodify nopeer notrap noquery** or add if not found.
5. Save and close the file.

For more information, please refer to:

https://support.ntp.org/bin/view/Support/AccessRestrictions

10.5 Secure Audit Trails

Secure audit trails so they cannot be altered.

10.5.2 Audit Trail Modification Protection

Protect audit trail files from unauthorized modifications.

10.5.2.1 Verify Audit Log Directories Mode

Verify Audit Log Directories Mode

Description	This test verifies that audit log directories have mode 0755 or less permissive. It prevents group and other users from deleting audit logs that audit trails can be modified or destroyed
Severity	0
Weight	5
Type	Content Test
Rules	Audit Log Directories Mode
Element	Equals "Audit Log Directories Mode"
Version conditions	If an element version has no content, the condition should:Pass Regular expression: /^[\ \t]*d.{4}-..-.\.?[\ \t].*/ (Flags:Multiline,Case insensitive,Comments mode) Audit Log Directories Have Mode 0755 Exists

Remediation

To remediate failure of this policy test, change the mode of the audit log directories to 755.

Changing the mode of the audit log directories:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   InitLogFile=`/bin/egrep "^[[:space:]]*log_file" /etc/audit/auditd.conf 2>/dev/null | /bin/sed s/^[^\/]*//`; DirName=`/usr/bin/dirname $InitLogFile 2>/dev/null`; /bin/ls -ldL $DirName | /bin/awk '{print $1,$3,$4,$NF}'
   ```

   to list the system audit log files' directory and their permissions.
3. Change permissions to **755** or less permissive using the **chmod g-w,o-w <*audit_log_directory*>** command.

For further details, please run the command **man chmod** to read man page.

10.5.2.2 Verify System Audit Log Files' Permission And Owner

Verify System Audit Log Files' Permission And Owner

Description

This test verifies that the 'root' user and the 'root' group own the system audit log files and permissions are set to 640 or more restrictive. Using permissions of 640 ensures that the 'root' user can not write, and the root group can not write and execute the file.

Severity

0

Weight

5

Type

Content Test

Rules

System Audit Log Permissions

Element

Equals "System Audit Log"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /^(?!-.{2}-.-{5}.?[\ \t]+root[\ \t]+root[\ \t]).*$/ (Flags:Multiline,Comments mode)
Invalid Audit Log Files Permission Does not exist

Remediation

To remediate failure of this policy test, change the mode to 640 or less permissive; change the owner and group-owner of the audit log files to root.

Changing the mode, owner and group-owner of the audit log files:

1. Become superuser or assume an equivalent role.
2. Run the script:

   ```
   InitLogFile=`/bin/egrep "^[[:space:]]*log_file" /etc/audit/auditd.conf 2>/dev/null | /bin/sed s/^[^\/]*//`; /bin/ls -ldL "$InitLogFile"* 2>/dev/null | /bin/awk '{print $1,$3,$4,$NF}'
   ```

   to list the system audit log files and its permissions.
3. Change permissions to 640 or more restrictive using the **chmod u-x,g-wx,o-rwx <*audit_logfile*>** command.
4. Change ownership using the **chown root:root <*audit_logfile*>** command.

For further details, please run the command **man chmod** and **man chown** to read man page.

10.5.2.3 Verify Log Files Permissions in /etc/rsyslog.conf

Verify Log Files Permissions in /etc/rsyslog.conf

Description

A log file must already exist for syslog to be able to write to it.
It is important to ensure that log files exist and have the correct permissions to ensure that sensitive syslog data is archived and protected.

Severity

0

Weight

5

Type

Content Test

Rules

Verify rsyslog Log Files Permissions

Element

Equals "Verify rsyslog Log Files Permissions"

Version conditions

If an element version has no content, the condition should:Pass
Regular expression: /.+/ (Flags:Case insensitive)
rsyslog Log Files Permissions Deviation Does not exist

Remediation

To remediate failure of this policy test, set appropriate permissions and ownership on the rsyslog log files.

Setting appropriate permissions and ownership on the rsyslog log files:

1. Become superuser or assume an equivalent role.
2. Using the following script to list all the rsyslog log files in the /etc/rsyslog.conf file:

```
/bin/awk -F "#" '$1 !~ /^[[:space:]]*\$/ && $1 !~ /\*[[:space:]]*$/
&& $1 !~ /^[[:space:]]*$/{ split($1,a," "); gsub(/-/,"",a[2]); if(a[
2] !~ /^@/ && a[2] ~ /^[[:space:]]*\//){ print a[2];} }' /etc/rsysl
og.conf 2>/dev/null
```

3. Run the command **touch <LOGFILE>** to create the files if they do not exist.
4. For sites that have not implemented a secure admin group, for each **<LOGFILE>** listed in the step 2, perform the following commands:

```
chown root:root <LOGFILE>
chmod u-x,og-rwx <LOGFILE>
```

5. For sites that have implemented a secure admin group, for each **<LOGFILE>** listed in the step 2, perform the following commands:

```
chown root:<SECURE_GROUP> <LOGFILE>
chmod u-x,g-wx,o-rwx <LOGFILE>
```

where **<SECURE_GROUP>** is the name of the security group.

10.7 Audit Trail Retention

Retain audit trail history for at least one year, with a minimum of three months immediately available for analysis (for example, online, archived, or restorable from back-up).

10.7.1 Verify That max_log_file_action Is Equal to keep_logs

Verify That max_log_file_action Is Equal to keep_logs

Description	Normally, auditd will hold 4 logs of maximum log file size before deleting older log files. In hight security contexts, the benefits of maintaining a long audit history exceed the cost of storing the audit history.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/audit/auditd.conf"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ ]*max_log_file_action[\ ]+=[\ ]+(\S+)[\ ]*$/ (Flags:Multiline,Case insensitive,Comments mode) max_log_file_action Value Matches "^(?:(?i)keep_logs)$"
Remediation	To remediate failure of this policy test, set the system action to take when the system has detected that the max file size limit has been reached. **Setting the system action to take when the system has detected that the max file size limit has been reached:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/audit/auditd.conf** file. 3. Find the line that contains **max_log_file_action = <value>**. 4. Set the **<value>** to **keep_logs** and save the file. 5. Run the **/usr/sbin/service auditd restart** command to apply the change. For further details, please run the command **man auditd.conf** to read man page.

Requirement 12 Maintain a Policy That Addresses Information Security for All Personnel

A strong security policy sets the security tone for the whole entity and informs personnel what is expected of them. All personnel should be aware of the sensitivity of data and their responsibilities for protecting it. For the purposes of Requirement 12, "personnel" refers to full-time and part-time employees, temporary employees, contractors and consultants who are "resident" on the entity's site or otherwise have access to the cardholder data environment.

12.3 Develop Technology Usage Policies

Develop usage policies for critical technologies (for example, remote-access technologies, wireless technologies, removable electronic media, laptops, tablets, personal data/digital assistants (PDAs), e-mail usage and Internet usage) and define proper use of these technologies. Ensure these usage policies require the following:

12.3.8 Automatic Session Disconnect

Automatic disconnect of sessions for remote-access technologies after a specific period of inactivity.

12.3.8.1 Verify That ClientAliveInterval Is Set to 900 or Less and Greater than 0

Verify That ClientAliveInterval Is Set to 900 or Less and Greater than 0

Description	The two options ClientAliveInterval and ClientAliveCountMax control the timeout of ssh sessions. When the ClientAliveInterval variable is set, ssh sessions that have no activity for the specified length of time are terminated. When the ClientAliveCountMax variable is set, sshd will send client alive messages at every ClientAliveInterval interval. When the number of consecutive client alive messages are sent with no response from the client, the ssh session is terminated. It is recommended that ClientAliveInterval is set to 900 (15 minutes) or less and greater than 0.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6 Red Hat Enterprise Linux Server 5
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*ClientAliveInterval[\ \t]+(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) ClientAliveInterval Timeout Less than or equal 900 AND ClientAliveInterval Timeout Greater than 0
Remediation	To remediate failure of this policy test, configure the SSH server to set a timeout interval in seconds after which if no data has been received from the client equals to 900 or less and greater than 0. **Configuring the SSH server to set a timeout interval in seconds after which if no data has been received from the client equals to 900 or less and greater than 0:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **ClientAliveInterval <value>**. 4. Set **<value>** to **900** or less and greater than **0** then save the file. 5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.

12.3.8.2 Verify That ClientAliveCountMax Is Set to 0

Verify That ClientAliveCountMax Is Set to 0

Description	This tests verifies that the SSH daemon is set a timeout count on idle sessions. This ensures a user login will be terminated as soon as the ClientAliveCountMax is reached. It is recommended that ClientAliveCountMax is set to 0.
Severity	0
Weight	5
Type	Content Test
Rules	System Configuration Files
Excluded Nodes	Red Hat Enterprise Linux Server 7 Red Hat Enterprise Linux Server 6
Element	Equals "/etc/ssh/sshd_config"
Version conditions	If an element version has no content, the condition should:Fail Regular expression: /^[\ \t]*ClientAliveCountMax[\ \t]+(\d+)[\ \t]*$/ (Flags:Multiline,Case insensitive,Comments mode) ClientAliveCountMax Equals 0
Remediation	To remediate failure of this policy test, configure the SSH server to set the number of client alive messages which may be sent without sshd receiving any messages back from the client equals to 0. **Configuring the SSH server to set the number of client alive messages which may be sent without sshd receiving any messages back from the client equals to 0:** 1. Become superuser or assume an equivalent role. 2. Open the **/etc/ssh/sshd_config** file. 3. Find the line **ClientAliveCountMax <value>**. 4. Set **<value>** to **0** and save the file. 5. Run the **pkill -HUP sshd** or **/sbin/service sshd restart** commands to restart the **sshd** service. For further details, please run the command **man sshd_config** to read man page.

Disclaimer

This remediation script is provided "AS IS" without any warranties of any kind. Tripwire is not responsible, and expressly disclaims all liability, for any modification of settings, undesired behavior or any other results of your use of this remediation. You assume all risk and responsibility therefore and you hereby agree to defend, indemnify and hold Tripwire harmless from any claims or damages related thereto. In any case, all modifications to systems should be performed by trained, experienced and appropriate IT staff. Always apply appropriate backup measures prior to configuration change to allow systems to be returned to prior state.