OpenSCAP Security Guide

Guide to the Secure Configuration of Red Hat Enterprise Linux 7

> with profile TAMU Red Hat Enterprise Linux 7 Benchmark Draft
>
> — This baseline draft aligns to the TAMU Controls Catalog for Red Hat Enterprise Linux 7.

The SCAP Security Guide Project
https://www.open-scap.org/security-policies/scap-security-guide
This guide presents a catalog of security-relevant configuration settings for Red Hat Enterprise Linux 7. It is a rendering of content structured in the eXtensible Configuration Checklist Description Format (XCCDF) in order to support security automation. The SCAP content is is available in the `scap-security-guide` package which is developed at https://www.open-scap.org/security-policies/scap-security-guide.

Providing system administrators with such guidance informs them how to securely configure systems under their control in a variety of network roles. Policy makers and baseline creators can use this catalog of settings, with its associated references to higher-level security control catalogs, in order to assist them in security baseline creation. This guide is a *catalog, not a checklist*, and satisfaction of every item is not likely to be possible or sensible in many operational scenarios. However, the XCCDF format enables granular selection and adjustment of settings, and their association with OVAL and OCIL content provides an automated checking capability. Transformations of this document, and its associated automated checking content, are capable of providing baselines that meet a diverse set of policy objectives. Some example XCCDF *Profiles*, which are selections of items that form checklists and can be used as baselines, are available with this guide. They can be processed, in an automated fashion, with tools that support the Security Content Automation Protocol (SCAP). The DISA STIG, which provides required settings for US Department of Defense systems, is one example of a baseline created from this guidance.

This benchmark is a direct port of a *SCAP Security Guide* benchmark developed for *Red Hat Enterprise Linux*. It has been modified through an automated process to remove specific dependencies on *Red Hat Enterprise Linux* and to function with *CentOS*. The result is a generally useful *SCAP Security Guide* benchmark with the following caveats:

- *CentOS* is not an exact copy of *Red Hat Enterprise Linux*. There may be configuration differences that produce false positives and/or false negatives. If this occurs please file a bug report.
- *CentOS* has its own build system, compiler options, patchsets, and is a community supported, non-commercial operating system. *CentOS* does not inherit certifications or evaluations from *Red Hat Enterprise Linux*. As such, some configuration rules (such as those requiring *FIPS 140-2* encryption) will continue to fail on *CentOS*.

Members of the *CentOS* community are invited to participate in OpenSCAP and SCAP Security Guide development. Bug reports and patches can be sent to GitHub: https://github.com/ComplianceAsCode/content. The mailing list is at https://fedorahosted.org/mailman/listinfo/scap-security-guide.

Do not attempt to implement any of the settings in this guide without first testing them in a non-operational environment. The creators of this guidance assume no responsibility whatsoever for its use by other parties, and makes no guarantees, expressed or implied, about its quality, reliability, or any other characteristic.

Profile Title	TAMU Red Hat Enterprise Linux 7 Benchmark Draft
Profile ID	xccdf_org.ssgproject.content_profile_tamu

Revision History

Current version: **0.1.59.2**

- **draft** (as of 2022-02-11)

Platforms

- cpe:/o:redhat:enterprise_linux:7
- cpe:/o:centos:centos:7
- cpe:/o:redhat:enterprise_linux:7::server
- cpe:/o:redhat:enterprise_linux:7::client
- cpe:/o:redhat:enterprise_linux:7::computenode
- cpe:/o:redhat:enterprise_linux:7::workstation

Table of Contents

Checklist

▼ contains 137 rules

System Settings [ref]

group

Contains rules that check correct system settings.

▼ contains 106 rules

Installing and Maintaining Software [ref]

group

The following sections contain information on security-relevant choices during the initial operating system installation process and the setup of software updates.

▼ contains 10 rules

System and Software Integrity [ref]

group

System and software integrity can be gained by installing antivirus, increasing system encryption strength with FIPS, verifying installed software, enabling SELinux, installing an Intrusion Prevention System, etc. However, installing or enabling integrity checking tools cannot *prevent* intrusions, but they can detect that an intrusion may have occurred. Requirements for integrity checking may be highly dependent on the environment in which the system will be used. Snapshot-based approaches such as AIDE may induce considerable overhead in the presence of frequent software updates.

▼ contains 2 rules

Software Integrity Checking [ref]

group

Both the AIDE (Advanced Intrusion Detection Environment) software and the RPM package management system provide mechanisms for verifying the integrity of installed software. AIDE uses snapshots of file metadata (such as hashes) and compares these to current system files in order to detect changes.

The RPM package management system can conduct integrity checks by comparing information in its metadata database with files installed on the system.

▼ contains 2 rules

Verify Integrity with RPM [ref]

group

The RPM package management system includes the ability to verify the integrity of installed packages by comparing the installed files with information about the files taken from the package metadata stored in the RPM database. Although an attacker could corrupt the RPM database (analogous to attacking the AIDE database as described above), this check can still reveal modification of important files. To list which files on the system differ from what is expected by the RPM database:

```
$ rpm -qVa
```

See the man page for `rpm` to see a complete explanation of each column.

▼ contains 2 rules

Verify File Hashes with RPM [ref]

rule

Without cryptographic integrity protections, system executables and files can be altered by unauthorized users without detection. The RPM package management system can check the hashes of installed software packages, including many that are important to system security. To verify that the cryptographic hash of system files and commands matches vendor values, run the following command to list which files on the system have hashes that differ from what is expected by the RPM database:

```
$ rpm -Va --noconfig | grep '^..5'
```

A "c" in the second column indicates that a file is a configuration file, which may appropriately be expected to change. If the file was not expected to change, investigate the cause of the change using audit logs or other means. The package can then be reinstalled to restore the file. Run the following command to determine which package owns the file:

```
$ rpm -qf FILENAME
```

The package can be reinstalled from a yum repository using the command:

```
$ sudo yum reinstall PACKAGENAME
```

Alternatively, the package can be reinstalled from trusted media using the command:

```
$ sudo rpm -Uvh PACKAGENAME
```

Rationale:
The hashes of important files like system executables should match the information given by the RPM database. Executables with erroneous hashes could be a sign of nefarious activity on the system.

Severity: high

References: 11, 2, 3, 9, 5.10.4.1, APO01.06, BAI03.05, BAI06.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS06.02, 3.3.8, 3.4.1, CCI-000366, CCI-001749, 164.308(a)(1)(ii)(D), 164.312(b), 164.312(c)(1), 164.312(c)(2), 164.312(e)(2)(i), 4.3.4.3.2, 4.3.4.3.3, 4.3.4.4.4, SR 3.1, SR 3.3, SR 3.4, SR 3.8, SR 7.6, A.11.2.4, A.12.1.2, A.12.2.1, A.12.5.1, A.12.6.2, A.14.1.2, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, CM-6(d), CM-6(c), SI-7, SI-7(1), SI-7(6), AU-9(3), PR.DS-6, PR.DS-8, PR.IP-1, Req-11.5, SRG-OS-000480-GPOS-00227, SV-214799r603261_rule, 6.1.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify and Correct File Permissions with RPM [ref] rule

The RPM package management system can check file access permissions of installed software packages, including many that are important to system security. Verify that the file permissions of system files and commands match vendor values. Check the file permissions with the following command:

```
$ sudo rpm -Va | awk '{ if (substr($0,2,1)=="M") print $NF }'
```

Output indicates files that do not match vendor defaults. After locating a file with incorrect permissions, run the following command to determine which package owns it:

```
$ rpm -qf FILENAME
```

Next, run the following command to reset its permissions to the correct values:

```
$ sudo rpm --setperms PACKAGENAME
```

Warning: Profiles may require that specific files have stricter file permissions than defined by the vendor. Such files will be reported as a finding and need to be evaluated according to your policy and deployment environment.

Rationale:
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: high

References: 1, 11, 12, 13, 14, 15, 16, 18, 3, 5, 6, 9, 5.10.4.1, APO01.06, APO11.04, BAI03.05, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.04, DSS05.07, DSS06.02, MEA02.01, 3.3.8, 3.4.1, CCI-001493, CCI-001494, CCI-001495, CCI-001496, 164.308(a)(1)(ii)(D), 164.312(b), 164.312(c)(1), 164.312(c)(2), 164.312(e)(2)(i), 4.3.3.3.9, 4.3.3.5.8, 4.3.3.7.3, 4.3.4.3.2, 4.3.4.3.3, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 2.1, SR 2.10, SR 2.11, SR 2.12, SR 2.8, SR 2.9, SR 5.2, SR 7.6, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.12.1.2, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.5.1, A.12.6.2, A.12.7.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R4.2, CIP-003-8 R6, CIP-007-3 R4, CIP-007-3 R4.1, CIP-007-3 R4.2, CM-6(d), CM-6(c), SI-7, SI-7(1), SI-7(6), AU-9(3), CM-6(a), PR.AC-4, PR.DS-5, PR.IP-1, PR.PT-1, Req-11.5, SRG-OS-000256-GPOS-00097, SRG-OS-000257-GPOS-00098, SRG-OS-000258-GPOS-00099, SRG-OS-000278-GPOS-00108, SV-204392r646841_rule, 1.7.1.4, 1.7.1.5, 1.7.1.6, 6.1.1, 6.1.2, 6.1.3, 6.1.4, 6.1.5, 6.1.6, 6.1.7, 6.1.8, 6.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disk Partitioning [ref]

group

To ensure separation and protection of data, there are top-level system directories which should be placed on their own physical partition or logical volume. The installer's default partitioning scheme creates separate logical volumes for `/`, `/boot`, and `swap`.

- If starting with any of the default layouts, check the box to \"Review and modify partitioning.\" This allows for the easy creation of additional logical volumes inside the volume group already created, though it may require making `/`'s logical volume smaller to create space. In general, using logical volumes is preferable to using partitions because they can be more easily adjusted later.
- If creating a custom layout, create the partitions mentioned in the previous paragraph (which the installer will require anyway), as well as separate ones described in the following sections.

If a system has already been installed, and the default partitioning scheme was used, it is possible but nontrivial to modify it to create separate logical volumes for the directories listed above. The Logical Volume Manager (LVM) makes this possible. See the LVM HOWTO at http://tldp.org/HOWTO/LVM-HOWTO/ for more detailed information on LVM.

▼ contains 2 rules

Ensure /var/log Located On Separate Partition [ref]

rule

System logs are stored in the `/var/log` directory. Ensure that `/var/log` has its own partition or logical volume at installation time, or migrate it using LVM.

Rationale:

Placing `/var/log` in its own partition enables better separation between log files and other files in `/var/`.

Severity: medium

References: BP28(R12), BP28(R47), 1, 12, 14, 15, 16, 3, 5, 6, 8, APO11.04, APO13.01, BAI03.05, DSS05.02, DSS05.04, DSS05.07, MEA02.01, CCI-000366, 4.3.3.3.9, 4.3.3.5.8, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 2.10, SR 2.11, SR 2.12, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, CIP-007-3 R6.5, CM-6(a), AU-4, SC-5(2), PR.PT-1, PR.PT-4, SRG-OS-000480-GPOS-00227, 1.1.15

Remediation Anaconda snippet: (show)

Remediation script: (show)

Ensure /var/log/audit Located On Separate Partition [ref]

rule

Audit logs are stored in the /var/log/audit directory. Ensure that /var/log/audit has its own partition or logical volume at installation time, or migrate it using LVM. Make absolutely certain that it is large enough to store all audit logs that will be created by the auditing daemon.

Rationale:
Placing /var/log/audit in its own partition enables better separation between audit files and other files, and helps ensure that auditing cannot be halted due to the partition running out of space.

Severity: low

References: BP28(R43), 1, 12, 13, 14, 15, 16, 2, 3, 5, 6, 8, APO11.04, APO13.01, BAI03.05, BAI04.04, DSS05.02, DSS05.04, DSS05.07, MEA02.01, CCI-000366, CCI-001849, 164.312(a)(2)(ii), 4.3.3.3.9, 4.3.3.5.8, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 2.10, SR 2.11, SR 2.12, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.2, SR 7.6, A.12.1.3, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.17.2.1, CIP-007-3 R6.5, CM-6(a), AU-4, SC-5(2), PR.DS-4, PR.PT-1, PR.PT-4, SRG-OS-000341-GPOS-00132, SRG-OS-000480-GPOS-00227, SRG-OS-000341-VMM-001220, SV-204495r603261_rule, 1.1.16

Remediation Anaconda snippet: (show)

Remediation script: (show)

Sudo [ref]

group

Sudo , which stands for "su 'do'", provides the ability to delegate authority to certain users, groups of users, or system administrators. When configured for system users and/or groups, Sudo can allow a user or group to execute privileged commands that normally only root is allowed to execute.

For more information on Sudo and addition Sudo configuration options, see **https://www.sudo.ws**.

▼ contains 3 rules

Install sudo Package [ref]

rule

The sudo package can be installed with the following command:

```
$ sudo yum install sudo
```

Rationale:
sudo is a program designed to allow a system administrator to give limited root privileges to users and log root activity. The basic philosophy is to give as few privileges as possible but still allow system users to get their work done.

Severity: medium

References: BP28(R19), 1382, 1384, 1386, CM-6(a), SRG-OS-000324-GPOS-00125, TAMU-AC-6(1), 5.2.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Remediation script: (show)

Process for least privilege [ref]

rule

Red Hat Enterprise Linux 7 requires elevation via `sudo` or `su` to grant elevated privileges.

Rationale:

From sudo's website https://www.sudo.ws/intro.html: Utilizing `sudo` to momentarily elevate an account's privileges allows a system administrator to give certain users (or groups of users) the ability to run some (or all) commands as root while logging all commands and arguments. Sudo operates on a per-command basis, it is not a replacement for the shell. Its features include:

- The ability to restrict what commands a user may run on a per-host basis.
- Sudo does copious logging of each command, providing a clear audit trail of who did what. When used in tandem with syslogd, the system log daemon, sudo can log all commands to a central host (as well as on the local host). All admins should use sudo in lieu of a root shell to take advantage of this logging.
- Sudo uses timestamp files to implement a ticketing system, of sorts. When a user invokes sudo and enters their password, they are granted a ticket for 5 minutes (this timeout is configurable at compile-time). Each subsequent sudo command updates the ticket for another 5 minutes. This avoids the problem of leaving a root shell where others can physically get to your keyboard. There is also an easy way for a user to remove their ticket file, useful for placing in a .logout file.
- Sudo's configuration file, the sudoers file, is setup in such a way that the same sudoers file may be used on many machines. This allows for central administration while keeping the flexibility to define a user's privileges on a per-host basis. Please see the samples sudoers file below for a real-world example

Severity: low

References: AC-6(1), AC-6(2), AC-6(5), AC-6(8), AC-6(9), TAMU-AC-6(all)

Ensure sudo Runs In A Minimal Environment - sudo env_reset [ref]

rule

The sudo `env_reset` tag, when specified, will run the command in a minimal environment, containing the TERM, PATH, HOME, MAIL, SHELL, LOGNAME, USER and SUDO_* variables. On Red Hat Enterprise Linux 7, `env_reset` is enabled by default This should be enabled by making sure that the `env_reset` tag exists in `/etc/sudoers` configuration file or any sudo configuration snippets in `/etc/sudoers.d/` .

Rationale:

Forcing sudo to reset the environment ensures that environment variables are not passed on to the command accidentaly, preventing leak of potentially sensitive information.

Severity: medium

References: BP28(R58), TAMU-AC-6(1)

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Updating Software [ref]

group

The `yum` command line tool is used to install and update software packages. The system also provides a graphical software update tool in the **System** menu, in the **Administration** submenu, called **Software Update**.

Red Hat Enterprise Linux 7 systems contain an installed software catalog called the RPM database, which records metadata of installed packages. Consistently using `yum` or the graphical **Software Update** for all software installation allows for insight into the current inventory of installed software on the system.

▼ contains 3 rules

Ensure gpgcheck Enabled In Main yum Configuration [ref]

rule

The gpgcheck option controls whether RPM packages' signatures are always checked prior to installation. To configure yum to check package signatures before installing them, ensure the following line appears in /etc/yum.conf in the [main] section:

```
gpgcheck=1
```

Rationale:

Changes to any software components can have significant effects on the overall security of the operating system. This requirement ensures the software has not been tampered with and that it has been provided by a trusted vendor.
Accordingly, patches, service packs, device drivers, or operating system components must be signed with a certificate recognized and approved by the organization.
Verifying the authenticity of the software prior to installation validates the integrity of the patch or upgrade received from a vendor. This ensures the software has not been tampered with and that it has been provided by a trusted vendor. Self-signed certificates are disallowed by this requirement. Certificates used to verify the software must be from an approved Certificate Authority (CA).

Severity: high

References: BP28(R15), 11, 2, 3, 9, 5.10.4.1, APO01.06, BAI03.05, BAI06.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS06.02, 3.4.8, CCI-001749, 164.308(a)(1)(ii)(D), 164.312(b), 164.312(c)(1), 164.312(c)(2), 164.312(e)(2)(i), 4.3.4.3.2, 4.3.4.3.3, 4.3.4.4.4, SR 3.1, SR 3.3, SR 3.4, SR 3.8, SR 7.6, A.11.2.4, A.12.1.2, A.12.2.1, A.12.5.1, A.12.6.2, A.14.1.2, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, CM-5(3), SI-7, SC-12, SC-12(3), CM-6(a), SA-12, SA-12(10), CM-11(a), CM-11(b), PR.DS-6, PR.DS-8, PR.IP-1, FPT_TUD_EXT.1, FPT_TUD_EXT.2, Req-6.2, SRG-OS-000366-GPOS-00153, SRG-OS-000366-VMM-001430, SRG-OS-000370-VMM-001460, SRG-OS-000404-VMM-001650, SV-204447r603261_rule, 1.2.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Operating System Version Supported [ref]

rule

By using LTS versions of Red Hat Enterprise Linux 7 the base OS is supported by the OS Vendor for a longer period of time.

Rationale:

Detailed information on OS release cycles, refer to the OS vendor. https://access.redhat.com/support/policy/updates/errata https://access.redhat.com/product-life-cycles?product=Red%20Hat%20Enterprise%20Linux,OpenShift%20Container%20Platform%204

Severity: low

References: SI-3(b), TAMU-SI-3(1.1)

Ensure Software Patches Installed [ref]

rule

If the system is joined to the Red Hat Network, a Red Hat Satellite Server, or a yum server, run the following command to install updates:

```
$ sudo yum update
```

If the system is not configured to use one of these sources, updates (in the form of RPM packages) can be manually downloaded from the Red Hat Network and installed using `rpm`.

NOTE: U.S. Defense systems are required to be patched within 30 days or sooner as local policy dictates.

Rationale:
Installing software updates is a fundamental mitigation against the exploitation of publicly-known vulnerabilities. If the most recent security patches and updates are not installed, unauthorized users may take advantage of weaknesses in the unpatched software. The lack of prompt attention to patching could result in a system compromise.

Severity: high

References: BP28(R08), 18, 20, 4, 5.10.4.1, APO12.01, APO12.02, APO12.03, APO12.04, BAI03.10, DSS05.01, DSS05.02, CCI-000366, CCI-001227, 4.2.3, 4.2.3.12, 4.2.3.7, 4.2.3.9, A.12.6.1, A.14.2.3, A.16.1.3, A.18.2.2, A.18.2.3, SI-2(5), SI-2(c), CM-6(a), ID.RA-1, PR.IP-12, FMT_MOF_EXT.1, Req-6.2, SRG-OS-000480-GPOS-00227, TAMU-CM-1(1), SRG-OS-000480-VMM-002000, SV-204459r603261_rule, 1.8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Account and Access Control [ref]

group

In traditional Unix security, if an attacker gains shell access to a certain login account, they can perform any action or access any file to which that account has access. Therefore, making it more difficult for unauthorized people to gain shell access to accounts, particularly to privileged accounts, is a necessary part of securing a system. This section introduces mechanisms for restricting access to accounts under Red Hat Enterprise Linux 7.

▼ contains 15 rules

Warning Banners for System Accesses [ref]

group

Each system should expose as little information about itself as possible.

System banners, which are typically displayed just before a login prompt, give out information about the service or the host's operating system. This might include the distribution name and the system kernel version, and the particular version of a network service. This information can assist intruders in gaining access to the system as it can reveal whether the system is running vulnerable software. Most network services can be configured to limit what information is displayed.

Many organizations implement security policies that require a system banner provide notice of the system's ownership, provide warning to unauthorized users, and remind authorized users of their consent to monitoring.

▼ contains 1 rule

Modify the System Login Banner [ref] rule

To configure the system login banner edit /etc/issue . Replace the default text with a message compliant with the local site policy or a legal disclaimer. The DoD required text is either:

You are accessing a U.S. Government (USG) Information System (IS) that is provided for USG-authorized use only. By using this IS (which includes any device attached to this IS), you consent to the following conditions:
-The USG routinely intercepts and monitors communications on this IS for purposes including, but not limited to, penetration testing, COMSEC monitoring, network operations and defense, personnel misconduct (PM), law enforcement (LE), and counterintelligence (CI) investigations.
-At any time, the USG may inspect and seize data stored on this IS.
-Communications using, or data stored on, this IS are not private, are subject to routine monitoring, interception, and search, and may be disclosed or used for any USG-authorized purpose.
-This IS includes security measures (e.g., authentication and access controls) to protect USG interests -- not for your personal benefit or privacy.
-Notwithstanding the above, using this IS does not constitute consent to PM, LE or CI investigative searching or monitoring of the content of privileged communications, or work product, related to personal representation or services by attorneys, psychotherapists, or clergy, and their assistants. Such communications and work product are private and confidential. See User Agreement for details.

OR:

I've read & consent to terms in IS user agreem't.

Rationale:
Display of a standardized and approved use notification before granting access to the operating system ensures privacy and security notification verbiage used is consistent with applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance.

System use notifications are required only for access via login interfaces with human users and are not required when such human interfaces do not exist.

Severity: medium

References: 1, 12, 15, 16, DSS05.04, DSS05.10, DSS06.10, 3.1.9, CCI-000048, CCI-000050, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, SR 1.1, SR 1.10, SR 1.2, SR 1.5, SR 1.7, SR 1.8, SR 1.9, A.18.1.4, A.9.2.1, A.9.2.4, A.9.3.1, A.9.4.2, A.9.4.3, AC-8(a), AC-8(c), PR.AC-7, FMT_MOF_EXT.1, SRG-OS-000023-GPOS-00006, SRG-OS-000024-GPOS-00007, TAMU-AC-8(2), SRG-OS-000023-VMM-000060, SRG-OS-000024-VMM-000070, SV-204395r603261_rule, 1.7.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Protect Accounts by Configuring PAM [ref] group

PAM, or Pluggable Authentication Modules, is a system which implements modular authentication for Linux programs. PAM provides a flexible and configurable architecture for authentication, and it should be configured to minimize exposure to unnecessary risk. This section contains guidance on how to accomplish that.

PAM is implemented as a set of shared objects which are loaded and invoked whenever an application wishes to authenticate a user. Typically, the application must be running as root in order to take advantage of PAM, because PAM's modules often need to be able to access sensitive stores of account information, such as /etc/shadow. Traditional privileged network listeners (e.g. sshd) or SUID programs (e.g. sudo) already meet this requirement. An SUID root application, userhelper, is provided so that programs which are not SUID or privileged themselves can still take advantage of PAM.

PAM looks in the directory `/etc/pam.d` for application-specific configuration information. For instance, if the program login attempts to authenticate a user, then PAM's libraries follow the instructions in the file `/etc/pam.d/login` to determine what actions should be taken.

One very important file in `/etc/pam.d` is `/etc/pam.d/system-auth`. This file, which is included by many other PAM configuration files, defines 'default' system authentication measures. Modifying this file is a good way to make far-reaching authentication changes, for instance when implementing a centralized authentication service.

Warning: Be careful when making changes to PAM's configuration files. The syntax for these files is complex, and modifications can have unexpected consequences. The default configurations shipped with applications should be sufficient for most users.

Warning: Running `authconfig` or `system-config-authentication` will re-write the PAM configuration files, destroying any manually made changes and replacing them with a series of system defaults. One reference to the configuration file syntax can be found at http://www.linux-pam.org/Linux-PAM-html/sag-configuration-file.html.

▼ contains 13 rules

Set Lockouts for Failed Password Attempts [ref] group

The `pam_faillock` PAM module provides the capability to lock out user accounts after a number of failed login attempts. Its documentation is available in `/usr/share/doc/pam-VERSION/txts/README.pam_faillock`.

Warning: Locking out user accounts presents the risk of a denial-of-service attack. The lockout policy must weigh whether the risk of such a denial-of-service attack outweighs the benefits of thwarting password guessing attacks.

▼ contains 4 rules

Limit Password Reuse [ref]

rule

Do not allow users to reuse recent passwords. This can be accomplished by using the remember option for the pam_unix or pam_pwhistory PAM modules.

In the file /etc/pam.d/system-auth, append remember=5 to the line which refers to the pam_unix.so or pam_pwhistory.so module, as shown below:

- for the pam_unix.so case:

```
password sufficient pam_unix.so ...existing_options... remember=5
```

- for the pam_pwhistory.so case:

```
password requisite pam_pwhistory.so ...existing_options... remember=5
```

The DoD STIG requirement is 5 passwords.

Rationale:
Preventing re-use of previous passwords helps ensure that a compromised password is not re-used by a user.

Severity: medium

References: BP28(R18), 1, 12, 15, 16, 5, 5.6.2.1.1, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.5.8, CCI-000200, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, A.18.1.4, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, IA-5(f), IA-5(1)(e), PR.AC-1, PR.AC-6, PR.AC-7, Req-8.2.5, SRG-OS-000077-GPOS-00045, SRG-OS-000077-VMM-000440, SV-204422r603261_rule, 5.4.4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Deny For Failed Password Attempts [ref]

rule

To configure the system to lock out accounts after a number of incorrect login attempts using pam_faillock.so, modify the content of both /etc/pam.d/system-auth and /etc/pam.d/password-auth as follows:

- add the following line immediately before the pam_unix.so statement in the AUTH section:

```
auth required pam_faillock.so preauth silent deny=3 unlock_time=900 fail_interval=900
```

- add the following line immediately after the pam_unix.so statement in the AUTH section:

```
auth [default=die] pam_faillock.so authfail deny=3 unlock_time=900 fail_interval=900
```

- add the following line immediately before the pam_unix.so statement in the ACCOUNT section:

```
account required pam_faillock.so
```

Rationale:
Locking out user accounts after a number of incorrect attempts prevents direct password guessing attacks.

Severity: medium

References: BP28(R18), 1, 12, 15, 16, 5.5.3, DSS05.04, DSS05.10, DSS06.10, 3.1.8, CCI-000044, CCI-002236, CCI-002237, CCI-002238, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, SR 1.1, SR 1.10, SR 1.2, SR 1.5, SR 1.7, SR 1.8, SR 1.9, 0421, 0422, 0431, 0974, 1173, 1401, 1504, 1505, 1546, 1557, 1558, 1559, 1560, 1561, A.18.1.4, A.9.2.1, A.9.2.4, A.9.3.1, A.9.4.2, A.9.4.3, CM-6(a), AC-7(a), PR.AC-7, FIA_AFL.1, Req-8.1.6, SRG-OS-000329-GPOS-00128, SRG-OS-000021-GPOS-00005, TAMU-AC-7(1), SRG-OS-000021-VMM-000050, SV-204427r603824_rule, 5.3.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Interval For Counting Failed Password Attempts [ref]

rule

Utilizing pam_faillock.so, the fail_interval directive configures the system to lock out an account after a number of incorrect login attempts within a specified time period. Modify the content of both /etc/pam.d/system-auth and /etc/pam.d/password-auth as follows:

- Add the following line immediately before the pam_unix.so statement in the AUTH section:

```
auth required pam_faillock.so preauth silent deny=3 unlock_time=900 fail_interval=900
```

- Add the following line immediately after the pam_unix.so statement in the AUTH section:

```
auth [default=die] pam_faillock.so authfail deny=3 unlock_time=900 fail_interval=900
```

- Add the following line immediately before the pam_unix.so statement in the ACCOUNT section:

```
account required pam_faillock.so
```

Rationale:
By limiting the number of failed logon attempts the risk of unauthorized system access via user password guessing, otherwise known as brute-forcing, is reduced. Limits are imposed by locking the account.

Severity: medium

References: BP28(R18), 1, 12, 15, 16, DSS05.04, DSS05.10, DSS06.10, CCI-000044, CCI-002236, CCI-002237, CCI-002238, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, SR 1.1, SR 1.10, SR 1.2, SR 1.5, SR 1.7, SR 1.8, SR 1.9, 0421, 0422, 0431, 0974, 1173, 1401, 1504, 1505, 1546, 1557, 1558, 1559, 1560, 1561, A.18.1.4, A.9.2.1, A.9.2.4, A.9.3.1, A.9.4.2, A.9.4.3, CM-6(a), AC-7(a), PR.AC-7, FIA_AFL.1, SRG-OS-000329-GPOS-00128, SRG-OS-000021-GPOS-00005, TAMU-AC-7(1), SRG-OS-000021-VMM-000050, SV-204427r603824_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Lockout Time for Failed Password Attempts [ref]

rule

To configure the system to lock out accounts after a number of incorrect login attempts and require an administrator to unlock the account using `pam_faillock.so`, modify the content of both `/etc/pam.d/system-auth` and `/etc/pam.d/password-auth` as follows:

- add the following line immediately `before` the `pam_unix.so` statement in the `AUTH` section:

```
auth required pam_faillock.so preauth silent deny=3 unlock_time=900 fail_interval=900
```

- add the following line immediately `after` the `pam_unix.so` statement in the `AUTH` section:

```
auth [default=die] pam_faillock.so authfail deny=3 unlock_time=900 fail_interval=900
```

- add the following line immediately `before` the `pam_unix.so` statement in the `ACCOUNT` section:

```
account required pam_faillock.so
```

If `unlock_time` is set to `0`, manual intervention by an administrator is required to unlock a user.

Rationale:
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: medium

References: BP28(R18), 1, 12, 15, 16, 5.5.3, DSS05.04, DSS05.10, DSS06.10, 3.1.8, CCI-000044, CCI-002236, CCI-002237, CCI-002238, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, SR 1.1, SR 1.10, SR 1.2, SR 1.5, SR 1.7, SR 1.8, SR 1.9, 0421, 0422, 0431, 0974, 1173, 1401, 1504, 1505, 1546, 1557, 1558, 1559, 1560, 1561, A.18.1.4, A.9.2.1, A.9.2.4, A.9.3.1, A.9.4.2, A.9.4.3, CM-6(a), AC-7(b), PR.AC-7, FIA_AFL.1, Req-8.1.7, SRG-OS-000329-GPOS-00128, SRG-OS-000021-GPOS-00005, TAMU-AC-7(1.1), SRG-OS-000329-VMM-001180, SV-204427r603824_rule, 5.3.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Password Quality Requirements [ref]

group

The default `pam_pwquality` PAM module provides strength checking for passwords. It performs a number of checks, such as making sure passwords are not similar to dictionary words, are of at least a certain length, are not the previous password reversed, and are not simply a change of case from the previous password. It can also require passwords to be in certain character classes. The `pam_pwquality` module is the preferred way of configuring password requirements.

The man pages `pam_pwquality(8)` provide information on the capabilities and configuration of each.

▼ contains 4 rules

Set Password Quality Requirements with pam_pwquality [ref] group

The pam_pwquality PAM module can be configured to meet requirements for a variety of policies.

For example, to configure pam_pwquality to require at least one uppercase character, lowercase character, digit, and other (special) character, make sure that pam_pwquality exists in /etc/pam.d/system-auth :

```
password    requisite     pam_pwquality.so try_first_pass local_users_only retry=3 authtok_type=
```

If no such line exists, add one as the first line of the password section in /etc/pam.d/system-auth . Next, modify the settings in /etc/security/pwquality.conf to match the following:

```
difok = 4
minlen = 14
dcredit = -1
ucredit = -1
lcredit = -1
ocredit = -1
maxrepeat = 3
```

The arguments can be modified to ensure compliance with your organization's security policy. Discussion of each parameter follows.

▼ contains 4 rules

Set Password Checks to Limit Use of GECOS Information [ref] rule

The pam_pwquality module's gecoscheck parameter controls requirements for not including identifiable information about the account. When set to a positive number, it will reject passwords which contain more than 3 characters from the passwd(5) GECOS field of the user. Modify the gecoscheck setting in /etc/security/pwquality.conf to be non-zero to prevent a run of 3 or more characters from the GECOS entry. The default is 0, which means that this check is disabled.

Rationale:

Use of a complex password helps to increase the time and resources required to compromise the password. Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks.

Password complexity is one factor of several that determines how long it takes to crack a password. The more complex the password, the greater the number of possible combinations that need to be tested before the password is compromised.

Passwords with personally identifiable information in them may be more vulnerable to password-guessing attacks.

Severity: medium

References: TAMU-IA-5(5.2), TAMU-IA-5(9.3.3)

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Password Maximum Consecutive Repeating Characters [ref]

rule

The pam_pwquality module's maxrepeat parameter controls requirements for consecutive repeating characters. When set to a positive number, it will reject passwords which contain more than that number of consecutive characters. Modify the maxrepeat setting in /etc/security/pwquality.conf to equal 1 to prevent a run of (1 + 1) or more identical characters.

Rationale:

Use of a complex password helps to increase the time and resources required to compromise the password. Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks.

Password complexity is one factor of several that determines how long it takes to crack a password. The more complex the password, the greater the number of possible combinations that need to be tested before the password is compromised.

Passwords with excessive repeating characters may be more vulnerable to password-guessing attacks.

Severity: medium

References: 1, 12, 15, 16, 5, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, CCI-000195, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, A.18.1.4, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, IA-5(c), CM-6(a), IA-5(4), PR.AC-1, PR.AC-6, PR.AC-7, SRG-OS-000072-GPOS-00040, TAMU-IA-5(5.2), TAMU-IA-5(9.3.5), SV-204413r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Password Maximum Consecutive Repeating Characters [ref]

rule

The pam_pwquality module's maxsequence parameter controls requirements for consecutive monotonic character sequences. When set to a positive number, it will reject passwords which contain more than that number of consecutive monotonic characters. Modify the maxsequence setting in /etc/security/pwquality.conf to equal 3 to prevent a run of (3 + 1) or more monotonic characters. The default is 0 which means that this check is disabled. Examples of such sequence are '12345' or 'fedcb'. Note that most such passwords will not pass the simplicity check unless the sequence is only a minor part of the password.

Rationale:

Use of a complex password helps to increase the time and resources required to compromise the password. Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks.

Password complexity is one factor of several that determines how long it takes to crack a password. The more complex the password, the greater the number of possible combinations that need to be tested before the password is compromised.

Passwords with excessive consequitive sequences of characters may be more vulnerable to password-guessing attacks.

Severity: medium

References: TAMU-IA-5(5.2), TAMU-IA-5(9.3.5)

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Ensure PAM Enforces Password Requirements - Minimum Length [ref] rule

The pam_pwquality module's minlen parameter controls requirements for minimum characters required in a password. Add minlen=20 after pam_pwquality to set minimum password length requirements.

Rationale:
The shorter the password, the lower the number of possible combinations that need to be tested before the password is compromised.
Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks. Password length is one factor of several that helps to determine strength and how long it takes to crack a password. Use of more characters in a password helps to exponentially increase the time and/or resources required to compromose the password.

Severity: medium

References: BP28(R18), 1, 12, 15, 16, 5, 5.6.2.1.1, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, CCI-000205, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, 0421, 0422, 0431, 0974, 1173, 1401, 1504, 1505, 1546, 1557, 1558, 1559, 1560, 1561, A.18.1.4, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, IA-5(c), IA-5(1)(a), CM-6(a), IA-5(4), PR.AC-1, PR.AC-6, PR.AC-7, FMT_MOF_EXT.1, Req-8.2.3, SRG-OS-000078-GPOS-00046, TAMU-IA-5(5.2), TAMU-IA-5(9.1), TAMU-IA-5(9.2), TAMU-IA-5(9.3.1), SRG-OS-000072-VMM-000390, SRG-OS-000078-VMM-000450, SV-204423r603261_rule, 5.4.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set Password Hashing Algorithm [ref] group

The system's default algorithm for storing password hashes in /etc/shadow is SHA-512. This can be configured in several locations.

▼ contains 3 rules

Set Password Hashing Algorithm in /etc/login.defs [ref]

rule

In /etc/login.defs , add or correct the following line to ensure the system will use SHA-512 as the hashing algorithm:

```
ENCRYPT_METHOD SHA512
```

Rationale:

Passwords need to be protected at all times, and encryption is the standard method for protecting passwords. If passwords are not encrypted, they can be plainly read (i.e., clear text) and easily compromised. Passwords that are encrypted with a weak algorithm are no more protected than if they are kept in plain text.

Using a stronger hashing algorithm makes password cracking attacks more difficult.

Severity: medium

References: BP28(R32), 1, 12, 15, 16, 5, 5.6.2.2, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.13.11, CCI-000196, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, 0418, 1055, 1402, A.18.1.4, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, IA-5(c), IA-5(1)(c), CM-6(a), PR.AC-1, PR.AC-6, PR.AC-7, Req-8.2.1, SRG-OS-000073-GPOS-00041, TAMU-IA-5(3.1), TAMU-IA-5(3.4), SV-204416r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set PAM's Password Hashing Algorithm [ref]

rule

The PAM system service can be configured to only store encrypted representations of passwords. In /etc/pam.d/system-auth , the password section of the file controls which PAM modules execute during a password change. Set the pam_unix.so module in the password section to include the argument sha512 , as shown below:

```
password    [success=1 default=ignore]    pam_unix.so sha512 other arguments...
```

This will help ensure when local users change their passwords, hashes for the new passwords will be generated using the SHA-512 algorithm. This is the default.

Rationale:

Passwords need to be protected at all times, and encryption is the standard method for protecting passwords. If passwords are not encrypted, they can be plainly read (i.e., clear text) and easily compromised. Passwords that are encrypted with a weak algorithm are no more protected than if they are kepy in plain text.

This setting ensures user and group account administration utilities are configured to store only encrypted representations of passwords. Additionally, the crypt_style configuration option ensures the use of a strong hashing algorithm that makes password cracking attacks more difficult.

Severity: medium

References: BP28(R32), 1, 12, 15, 16, 5, 5.6.2.2, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.13.11, CCI-000196, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, 0418, 1055, 1402, A.18.1.4, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, IA-5(c), IA-5(1)(c), CM-6(a), PR.AC-1, PR.AC-6, PR.AC-7, Req-8.2.1, SRG-OS-000073-GPOS-00041, TAMU-IA-5(3.1), TAMU-IA-5(3.4), SRG-OS-000480-VMM-002000, SV-204415r603261_rule, 5.4.3

Remediation Shell script: (show)

Set Password Hashing Rounds in /etc/login.defs [ref]

rule

In /etc/login.defs , ensure SHA_CRYPT_MIN_ROUNDS and SHA_CRYPT_MAX_ROUNDS has the minimum value of 5000 . For example:

```
SHA_CRYPT_MIN_ROUNDS 5000
SHA_CRYPT_MAX_ROUNDS 5000
```

Notice that if neither are set, they already have the default value of 5000. If either is set, they must have the minimum value of 5000.

Rationale:
Passwords need to be protected at all times, and encryption is the standard method for protecting passwords. If passwords are not encrypted, they can be plainly read (i.e., clear text) and easily compromised. Passwords that are encrypted with a weak algorithm are no more protected than if they are kept in plain text.

Using more hashing rounds makes password cracking attacks more difficult.

Severity: medium

References: CCI-000803, SRG-OS-000073-GPOS-00041, SRG-OS-000120-GPOS-00061, TAMU-IA-5(3.1), TAMU-IA-5(3.4)

Remediation Ansible snippet: (show)

Password Failure Message [ref]

rule

Red Hat Enterprise Linux 7 natively does not disclose to the login screen the reasons for a login failure. The failed login output says "Permission denied" only; no indication as to whether the username or password was wrong.

Rationale:
Disclosing that an account exists can be just as useful as an attack vector as determining if the password entered was correct.

Severity: low

References: TAMU-IA-6(1.2)

Password Display on Entry [ref]

rule

Red Hat Enterprise Linux 7 natively obscures the typing feedback of authentication information during password entry. No feedback is given while passwords are being typed.

Rationale:
The feedback from information systems does not provide information that would allow unauthorized individuals to compromise authentication mechanisms. For some types of information systems or system components, for example, desktops/notebooks with relatively large monitors, the threat (often referred to as shoulder surfing) may be significant. For other types of systems or components, for example, mobile devices with 2-4 inch screens, this threat may be less significant, and may need to be balanced against the increased likelihood of typographic input errors due to the small keyboards. Therefore, the means for obscuring the authenticator feedback is selected accordingly. Obscuring the feedback of authentication information includes, for example, displaying asterisks when users type passwords into input devices, or displaying feedback for a very limited time before fully obscuring it.

Severity: low

References: IA-6, TAMU-IA-6(1.1)

Protect Accounts by Restricting Password-Based Login [ref] group

Conventionally, Unix shell accounts are accessed by providing a username and password to a login program, which tests these values for correctness using the /etc/passwd and /etc/shadow files. Password-based login is vulnerable to guessing of weak passwords, and to sniffing and man-in-the-middle attacks against passwords entered over a network or at an insecure console. Therefore, mechanisms for accessing accounts by entering usernames and passwords should be restricted to those which are operationally necessary.

▼ contains 1 rule

Verify Proper Storage and Existence of Password Hashes [ref] group

By default, password hashes for local accounts are stored in the second field (colon-separated) in /etc/shadow. This file should be readable only by processes running with root credentials, preventing users from casually accessing others' password hashes and attempting to crack them. However, it remains possible to misconfigure the system and store password hashes in world-readable files such as /etc/passwd, or to even store passwords themselves in plaintext on the system. Using system-provided tools for password change/creation should allow administrators to avoid such misconfiguration.

▼ contains 1 rule

Prevent Login to Accounts With Empty Password [ref] rule

If an account is configured for password authentication but does not have an assigned password, it may be possible to log into the account without authentication. Remove any instances of the nullok in /etc/pam.d/system-auth to prevent logins with empty passwords. Note that this rule is not applicable for systems running within a container. Having user with empty password within a container is not considered a risk, because it should not be possible to directly login into a container anyway.

Rationale:
If an account has an empty password, anyone could log in and run commands with the privileges of that account. Accounts with empty passwords should never be used in operational environments.

Severity: high

References: 1, 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2, APO01.06, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.02, DSS06.03, DSS06.10, 3.1.1, 3.1.5, CCI-000366, 164.308(a)(1)(ii)(B), 164.308(a)(7)(i), 164.308(a)(7)(ii)(A), 164.310(a)(1), 164.310(a)(2)(i), 164.310(a)(2)(ii), 164.310(a)(2)(iii), 164.310(b), 164.310(c), 164.310(d)(1), 164.310(d)(2)(iii), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.18.1.4, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.1, A.9.4.2, A.9.4.3, A.9.4.4, A.9.4.5, IA-5(1)(a), IA-5(c), CM-6(a), PR.AC-1, PR.AC-4, PR.AC-6, PR.AC-7, PR.DS-5, FIA_UAU.1, Req-8.2.3, SRG-OS-000480-GPOS-00227, SV-204424r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

System Accounting with auditd [ref] group

The audit service provides substantial capabilities for recording system activities. By default, the service audits about SELinux AVC denials and certain types of security-relevant events such as system logins, account modifications, and authentication events performed by programs such as sudo. Under its default configuration, auditd has modest disk space requirements, and should not noticeably impact system performance.

NOTE: The Linux Audit daemon `auditd` can be configured to use the `augenrules` program to read audit rules files (`*.rules`) located in `/etc/audit/rules.d` location and compile them to create the resulting form of the `/etc/audit/audit.rules` configuration file during the daemon startup (default configuration). Alternatively, the `auditd` daemon can use the `auditctl` utility to read audit rules from the `/etc/audit/audit.rules` configuration file during daemon startup, and load them into the kernel. The expected behavior is configured via the appropriate `ExecStartPost` directive setting in the `/usr/lib/systemd/system/auditd.service` configuration file. To instruct the `auditd` daemon to use the `augenrules` program to read audit rules (default configuration), use the following setting:

```
ExecStartPost=-/sbin/augenrules --load
```

in the `/usr/lib/systemd/system/auditd.service` configuration file. In order to instruct the `auditd` daemon to use the `auditctl` utility to read audit rules, use the following setting:

```
ExecStartPost=-/sbin/auditctl -R /etc/audit/audit.rules
```

in the `/usr/lib/systemd/system/auditd.service` configuration file. Refer to `[Service]` section of the `/usr/lib/systemd/system/auditd.service` configuration file for further details.

Government networks often have substantial auditing requirements and `auditd` can be configured to meet these requirements. Examining some example audit records demonstrates how the Linux audit system satisfies common requirements. The following example from Fedora Documentation available at https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html-single/selinux_users_and_administrators_guide/index#sect-Security-Enhanced_Linux-Fixing_Problems-Raw_Audit_Messages shows the substantial amount of information captured in a two typical "raw" audit messages, followed by a breakdown of the most important fields. In this example the message is SELinux-related and reports an AVC denial (and the associated system call) that occurred when the Apache HTTP Server attempted to access the `/var/www/html/file1` file (labeled with the `samba_share_t` type):

```
type=AVC msg=audit(1226874073.147:96): avc:  denied  { getattr } for pid=2465 comm="httpd"
path="/var/www/html/file1" dev=dm-0 ino=284133 scontext=unconfined_u:system_r:httpd_t:s0
tcontext=unconfined_u:object_r:samba_share_t:s0 tclass=file

type=SYSCALL msg=audit(1226874073.147:96): arch=40000003 syscall=196 success=no exit=-13
a0=b98df198 a1=bfec85dc a2=54dff4 a3=2008171 items=0 ppid=2463 pid=2465 auid=502 uid=48
gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=6 comm="httpd"
exe="/usr/sbin/httpd" subj=unconfined_u:system_r:httpd_t:s0 key=(null)
```

- `msg=audit(1226874073.147:96)`
  - The number in parentheses is the unformatted time stamp (Epoch time) for the event, which can be converted to standard time by using the `date` command.
- `{ getattr }`
  - The item in braces indicates the permission that was denied. `getattr` indicates the source process was trying to read the target file's status information. This occurs before reading files. This action is denied due to the file being accessed having the wrong label. Commonly seen permissions include `getattr` , `read` , and `write` .
- `comm="httpd"`
  - The executable that launched the process. The full path of the executable is found in the `exe=` section of the system call (`SYSCALL`) message, which in this case, is `exe="/usr/sbin/httpd"` .
- `path="/var/www/html/file1"`
  - The path to the object (target) the process attempted to access.
- `scontext="unconfined_u:system_r:httpd_t:s0"`
  - The SELinux context of the process that attempted the denied action. In this case, it is the SELinux context of the Apache HTTP Server, which is running in the `httpd_t` domain.
- `tcontext="unconfined_u:object_r:samba_share_t:s0"`
  - The SELinux context of the object (target) the process attempted to access. In this case, it is the SELinux context of `file1` . Note: the `samba_share_t` type is not accessible to processes running in the `httpd_t` domain.
- From the system call (`SYSCALL`) message, two items are of interest:
  - `success=no` : indicates whether the denial (AVC) was enforced or not. `success=no` indicates the system call was not successful (SELinux denied access). `success=yes` indicates the system call was successful - this can be seen for permissive domains or unconfined domains, such as `initrc_t` and `kernel_t` .
  - `exe="/usr/sbin/httpd"` : the full path to the executable that launched the process, which in this case, is `exe="/usr/sbin/httpd"` .

▼ contains 29 rules

Configure auditd Rules for Comprehensive Auditing [ref]

group

The auditd program can perform comprehensive monitoring of system activity. This section describes recommended configuration settings for comprehensive auditing, but a full description of the auditing system's capabilities is beyond the scope of this guide. The mailing list *linux-audit@redhat.com* exists to facilitate community discussion of the auditing system.

The audit subsystem supports extensive collection of events, including:

- Tracing of arbitrary system calls (identified by name or number) on entry or exit.
- Filtering by PID, UID, call success, system call argument (with some limitations), etc.
- Monitoring of specific files for modifications to the file's contents or metadata.

Auditing rules at startup are controlled by the file /etc/audit/audit.rules. Add rules to it to meet the auditing requirements for your organization. Each line in /etc/audit/audit.rules represents a series of arguments that can be passed to auditctl and can be individually tested during runtime. See documentation in /usr/share/doc/audit-*VERSION* and in the related man pages for more details.

If copying any example audit rulesets from /usr/share/doc/audit-VERSION, be sure to comment out the lines containing arch= which are not appropriate for your system's architecture. Then review and understand the following rules, ensuring rules are activated as needed for the appropriate architecture.

After reviewing all the rules, reading the following sections, and editing as needed, the new rules can be activated as follows:

```
$ sudo service auditd restart
```

▼ contains 27 rules

Record Events that Modify the System's Discretionary Access Controls [ref]

group

At a minimum, the audit system should collect file permission changes for all users and root. Note that the "-F arch=b32" lines should be present even on a 64 bit system. These commands identify system calls for auditing. Even if the system is 64 bit it can still execute 32 bit system calls. Additionally, these rules can be configured in a number of ways while still achieving the desired effect. An example of this is that the "-S" calls could be split up and placed on separate lines, however, this is less efficient. Add the following to /etc/audit/audit.rules:

```
-a always,exit -F arch=b32 -S chmod,fchmod,fchmodat -F auid>=1000 -F auid!=unset -F key=perm_mod
    -a always,exit -F arch=b32 -S chown,fchown,fchownat,lchown -F auid>=1000 -F auid!=unset -F key=perm_mod
    -a always,exit -F arch=b32 -S setxattr,lsetxattr,fsetxattr,removexattr,lremovexattr,fremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If your system is 64 bit then these lines should be duplicated and the arch=b32 replaced with arch=b64 as follows:

```
-a always,exit -F arch=b64 -S chmod,fchmod,fchmodat -F auid>=1000 -F auid!=unset -F key=perm_mod
    -a always,exit -F arch=b64 -S chown,fchown,fchownat,lchown -F auid>=1000 -F auid!=unset -F key=perm_mod
    -a always,exit -F arch=b64 -S setxattr,lsetxattr,fsetxattr,removexattr,lremovexattr,fremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

▼ contains 13 rules

Record Events that Modify the System's Discretionary Access Controls - chmod [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S chmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S chmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S chmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S chmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204521r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - chown [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d` :

```
-a always,exit -F arch=b32 -S chown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S chown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S chown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S chown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000474-GPOS-00219, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204517r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - fchmod [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d`:

```
-a always,exit -F arch=b32 -S fchmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S fchmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchmod -F auid>=1000 -F auid!=unset -F key=perm_mod
```

> **Warning:** Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204522r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - fchmodat [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S fchmodat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchmodat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S fchmodat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchmodat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204523r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - fchown [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S fchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S fchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000474-GPOS-00219, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204518r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - fchownat [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S fchownat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchownat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S fchownat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fchownat -F auid>=1000 -F auid!=unset -F key=perm_mod
```

> **Warning:** Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000474-GPOS-00219, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204520r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - fremovexattr [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root.

If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d` :

```
-a always,exit -F arch=b32 -S fremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S fremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:

The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000458-GPOS-00203, SRG-OS-000462-GPOS-00206, SRG-OS-000463-GPOS-00207, SRG-OS-000471-GPOS-00215, SRG-OS-000474-GPOS-00219, SRG-OS-000466-GPOS-00210, SRG-OS-000064-GPOS-00033, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204528r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - fsetxattr [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d` :

```
-a always,exit -F arch=b32 -S fsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S fsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S fsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000458-GPOS-00203, SRG-OS-000462-GPOS-00206, SRG-OS-000463-GPOS-00207, SRG-OS-000468-GPOS-00212, SRG-OS-000471-GPOS-00215, SRG-OS-000474-GPOS-00219, SRG-OS-000064-GPOS-00033, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204525r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - lchown [ref]

rule

At a minimum, the audit system should collect file permission changes for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d`:

```
-a always,exit -F arch=b32 -S lchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S lchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S lchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S lchown -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:

The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000466-GPOS-00210, SRG-OS-000458-GPOS-00203, SRG-OS-000474-GPOS-00219, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204519r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - lremovexattr [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root.

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S lremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S lremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S lremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S lremovexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000458-GPOS-00203, SRG-OS-000462-GPOS-00206, SRG-OS-000463-GPOS-00207, SRG-OS-000468-GPOS-00212, SRG-OS-000471-GPOS-00215, SRG-OS-000474-GPOS-00219, SRG-OS-000466-GPOS-00210, SRG-OS-000064-GPOS-00033, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204529r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - lsetxattr [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d` :

```
-a always,exit -F arch=b32 -S lsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S lsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S lsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S lsetxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000458-GPOS-00203, SRG-OS-000462-GPOS-00206, SRG-OS-000463-GPOS-00207, SRG-OS-000468-GPOS-00212, SRG-OS-000471-GPOS-00215, SRG-OS-000474-GPOS-00219, SRG-OS-000064-GPOS-00033, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204526r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - removexattr [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root.

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S removexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S removexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S removexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S removexattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:

The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000458-GPOS-00203, SRG-OS-000462-GPOS-00206, SRG-OS-000463-GPOS-00207, SRG-OS-000468-GPOS-00212, SRG-OS-000471-GPOS-00215, SRG-OS-000474-GPOS-00219, SRG-OS-000466-GPOS-00210, SRG-OS-000064-GPOS-00033, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204527r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Discretionary Access Controls - setxattr [ref] rule

At a minimum, the audit system should collect file permission changes for all users and root. If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S setxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S setxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S setxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S setxattr -F auid>=1000 -F auid!=unset -F key=perm_mod
```

Warning: Note that these rules can be configured in a number of ways while still achieving the desired effect. Here the system calls have been placed independent of other system calls. Grouping these system calls with others as identifying earlier in this guide is more efficient.

Rationale:
The changing of file permissions could indicate that a user is attempting to gain access to information that would otherwise be disallowed. Auditing DAC modifications can facilitate the identification of patterns of abuse among both authorized and unauthorized users.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000064-GPOS-00033, SRG-OS-000458-GPOS-00203, SRG-OS-000458-VMM-001810, SRG-OS-000474-VMM-001940, SV-204524r603261_rule, 4.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record File Deletion Events by User [ref]

group

At a minimum, the audit system should collect file deletion events for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d`, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S rmdir,unlink,unlinkat,rename,renameat -F auid>=1000 -F auid!=unset -F key=delete
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S rmdir,unlink,unlinkat,rename,renameat -F auid>=1000 -F auid!=unset -F key=delete
```

▼ contains 1 rule

Ensure auditd Collects File Deletion Events by User [ref]

rule

At a minimum the audit system should collect file deletion events for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d`, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S rmdir,unlink,unlinkat,rename,renameat -F auid>=1000 -F auid!=unset -F key=delete
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S rmdir,unlink,unlinkat,rename -S renameat -F auid>=1000 -F auid!=unset -F key=delete
```

Warning: This rule checks for multiple syscalls related to file deletion; it was written with DISA STIG in mind. Other policies should use a separate rule for each syscall that needs to be checked. For example:

- `audit_rules_file_deletion_events_rmdir`
- `audit_rules_file_deletion_events_unlink`
- `audit_rules_file_deletion_events_unlinkat`

Rationale:

Auditing file deletions will create an audit trail for files that are removed from the system. The audit trail could aid in system troubleshooting, as well as, detecting malicious processes that attempt to delete log files to conceal their presence.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000366, CCI-000172, CCI-002884, 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.2.7, 4.1.14

Remediation Shell script: (show)

Record Unauthorized Access Attempts Events to Files (unsuccessful) [ref]

group

At a minimum, the audit system should collect unauthorized file accesses for all users and root. Note that the "-F arch=b32" lines should be present even on a 64 bit system. These commands identify system calls for auditing. Even if the system is 64 bit it can still execute 32 bit system calls. Additionally, these rules can be configured in a number of ways while still achieving the desired effect. An example of this is that the "-S" calls could be split up and placed on separate lines, however, this is less efficient. Add the following to `/etc/audit/audit.rules`:

```
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -F key=access
    -a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -F key=access
```

If your system is 64 bit then these lines should be duplicated and the arch=b32 replaced with arch=b64 as follows:

```
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -F key=access
    -a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -F key=access
```

▼ contains 1 rule

Ensure auditd Collects Unauthorized Access Attempts to Files (unsuccessful) [ref] rule

At a minimum the audit system should collect unauthorized file accesses for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following lines to a file with suffix `.rules` in the directory `/etc/audit/rules.d`:

```
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -F key=access
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -F key=access
```

If the system is 64 bit then also add the following lines:

```
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -F key=access
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -F key=access
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following lines to `/etc/audit/audit.rules` file:

```
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -F key=access
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -F key=access
```

If the system is 64 bit then also add the following lines:

```
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -F key=access
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -F key=access
```

Warning: This rule checks for multiple syscalls related to unsuccessful file modification; it was written with DISA STIG in mind. Other policies should use a separate rule for each syscall that needs to be checked. For example:

- `audit_rules_unsuccessful_file_modification_open`
- `audit_rules_unsuccessful_file_modification_ftruncate`
- `audit_rules_unsuccessful_file_modification_creat`

Rationale:

Unsuccessful attempts to access files could be an indicator of malicious activity on a system. Auditing these events could serve as evidence of potential system compromise.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000172, CCI-002884, 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, 0582, 0584, 05885, 0586, 0846, 0957, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.2.4, Req-10.2.1

Remediation Shell script: (show)

Record Information on Kernel Modules Loading and Unloading [ref] group

To capture kernel module loading and unloading events, use following lines, setting ARCH to either b32 for 32-bit system, or having two lines for both b32 and b64 in case your system is 64-bit:

```
-a always,exit -F arch=ARCH -S init_module,delete_module -F key=modules
```

Place to add the lines depends on a way `auditd` daemon is configured. If it is configured to use the `augenrules` program (the default), add the lines to a file with suffix `.rules` in the directory `/etc/audit/rules.d` . If the `auditd` daemon is configured to use the `auditctl` utility, add the lines to file `/etc/audit/audit.rules` .

▼ contains 1 rule

Ensure auditd Collects Information on Kernel Module Loading and Unloading [ref] rule

To capture kernel module loading and unloading events, use following lines, setting ARCH to either b32 for 32-bit system, or having two lines for both b32 and b64 in case your system is 64-bit:

```
-a always,exit -F arch=ARCH -S init_module,finit_module,delete_module -F key=modules
```

The place to add the lines depends on a way `auditd` daemon is configured. If it is configured to use the `augenrules` program (the default), add the lines to a file with suffix `.rules` in the directory `/etc/audit/rules.d` . If the `auditd` daemon is configured to use the `auditctl` utility, add the lines to file `/etc/audit/audit.rules` .

Rationale:

The addition/removal of kernel modules can be used to alter the behavior of the kernel and potentially introduce malicious code into kernel space. It is important to have an audit trail of modules that have been introduced into the kernel.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000172, 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.2.7, 4.1.17

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Information on the Use of Privileged Commands [ref] group

At a minimum, the audit system should collect the execution of privileged commands for all users and root.

▼ contains 1 rule

Ensure auditd Collects Information on the Use of Privileged Commands [ref] rule

The audit system should collect information about usage of privileged commands for all users and root. To find the relevant setuid / setgid programs, run the following command for each local partition *PART*:

```
$ sudo find PART -xdev -type f -perm -4000 -o -type f -perm -2000 2>/dev/null
```

If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add a line of the following form to a file with suffix `.rules` in the directory `/etc/audit/rules.d` for each setuid / setgid program on the system, replacing the *SETUID_PROG_PATH* part with the full path of that setuid / setgid program in the list:

```
-a always,exit -F path=SETUID_PROG_PATH -F auid>=1000 -F auid!=unset -F key=privileged
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add a line of the following form to `/etc/audit/audit.rules` for each setuid / setgid program on the system, replacing the *SETUID_PROG_PATH* part with the full path of that setuid / setgid program in the list:

```
-a always,exit -F path=SETUID_PROG_PATH -F auid>=1000 -F auid!=unset -F key=privileged
```

Warning: This rule checks for multiple syscalls related to privileged commands; it was written with DISA STIG in mind. Other policies should use a separate rule for each syscall that needs to be checked. For example:

- `audit_rules_privileged_commands_su`
- `audit_rules_privileged_commands_umount`
- `audit_rules_privileged_commands_passwd`

Rationale:

Misuse of privileged functions, either intentionally or unintentionally by authorized users, or by unauthorized external entities that have compromised system accounts, is a serious and ongoing concern and can have significant adverse impacts on organizations. Auditing the use of privileged functions is one way to detect such misuse and identify the risk from insider and advanced persistent threats.

Privileged programs are subject to escalation-of-privilege attacks, which attempt to subvert their normal role of providing some necessary but limited capability. As such, motivation exists to monitor these programs for unusual activity.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO08.04, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.05, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-002234, 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.5, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.3.4.5.9, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 3.9, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, 0582, 0584, 05885, 0586, 0846, 0957, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.1, A.16.1.2, A.16.1.3, A.16.1.4, A.16.1.5, A.16.1.7, A.6.1.3, A.6.2.1, A.6.2.2, CIP-004-6 R2.2.2, CIP-004-6 R2.2.3, CIP-007-3 R.1.3, CIP-007-3 R5, CIP-007-3 R5.1.1, CIP-007-3 R5.1.3, CIP-007-3 R5.2.1, CIP-007-3 R5.2.3, AC-2(4), AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-2, DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, DE.DP-4, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, RS.CO-2, Req-10.2.2, SRG-OS-000327-GPOS-00127, SRG-OS-000471-VMM-001910, 4.1.12

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Records Events that Modify Date and Time Information [ref] group

Arbitrary changes to the system time can be used to obfuscate nefarious activities in log files, as well as to confuse network services that are highly dependent upon an accurate system time. All changes to the system time should be audited.

contains 5 rules

Record attempts to alter time through adjtimex [ref]

rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S adjtimex -F key=audit_time_rules
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S adjtimex -F key=audit_time_rules
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S adjtimex -F key=audit_time_rules
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S adjtimex -F key=audit_time_rules
```

The -k option allows for the specification of a key in string form that can be used for better reporting capability through ausearch and aureport. Multiple system calls can be defined on the same line to save space if desired, but is not required. See an example of multiple combined syscalls:

```
-a always,exit -F arch=b64 -S adjtimex,settimeofday -F key=audit_time_rules
```

Rationale:

Arbitrary changes to the system time can be used to obfuscate nefarious activities in log files, as well as to confuse network services that are highly dependent upon an accurate system time (such as sshd). All changes to the system time should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-001487, CCI-000169, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.4.2.b, 4.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Record Attempts to Alter Time Through clock_settime [ref] rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S clock_settime -F a0=0x0 -F key=time-change
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S clock_settime -F a0=0x0 -F key=time-change
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S clock_settime -F a0=0x0 -F key=time-change
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S clock_settime -F a0=0x0 -F key=time-change
```

The -k option allows for the specification of a key in string form that can be used for better reporting capability through ausearch and aureport. Multiple system calls can be defined on the same line to save space if desired, but is not required. See an example of multiple combined syscalls:

```
-a always,exit -F arch=b64 -S adjtimex,settimeofday -F key=audit_time_rules
```

Rationale:
Arbitrary changes to the system time can be used to obfuscate nefarious activities in log files, as well as to confuse network services that are highly dependent upon an accurate system time (such as sshd). All changes to the system time should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-001487, CCI-000169, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.4.2.b, 4.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Record attempts to alter time through settimeofday [ref]

rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-a always,exit -F arch=b32 -S settimeofday -F key=audit_time_rules
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S settimeofday -F key=audit_time_rules
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-a always,exit -F arch=b32 -S settimeofday -F key=audit_time_rules
```

If the system is 64 bit then also add the following line:

```
-a always,exit -F arch=b64 -S settimeofday -F key=audit_time_rules
```

The -k option allows for the specification of a key in string form that can be used for better reporting capability through ausearch and aureport. Multiple system calls can be defined on the same line to save space if desired, but is not required. See an example of multiple combined syscalls:

```
-a always,exit -F arch=b64 -S adjtimex,settimeofday -F key=audit_time_rules
```

Rationale:
Arbitrary changes to the system time can be used to obfuscate nefarious activities in log files, as well as to confuse network services that are highly dependent upon an accurate system time (such as sshd). All changes to the system time should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-001487, CCI-000169, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.4.2.b, 4.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Record Attempts to Alter Time Through stime [ref] rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d for both 32 bit and 64 bit systems:

```
-a always,exit -F arch=b32 -S stime -F key=audit_time_rules
```

Since the 64 bit version of the "stime" system call is not defined in the audit lookup table, the corresponding "-F arch=b64" form of this rule is not expected to be defined on 64 bit systems (the aforementioned "-F arch=b32" stime rule form itself is sufficient for both 32 bit and 64 bit systems). If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file for both 32 bit and 64 bit systems:

```
-a always,exit -F arch=b32 -S stime -F key=audit_time_rules
```

Since the 64 bit version of the "stime" system call is not defined in the audit lookup table, the corresponding "-F arch=b64" form of this rule is not expected to be defined on 64 bit systems (the aforementioned "-F arch=b32" stime rule form itself is sufficient for both 32 bit and 64 bit systems). The -k option allows for the specification of a key in string form that can be used for better reporting capability through ausearch and aureport. Multiple system calls can be defined on the same line to save space if desired, but is not required. See an example of multiple combined system calls:

```
-a always,exit -F arch=b64 -S adjtimex,settimeofday -F key=audit_time_rules
```

Rationale:
Arbitrary changes to the system time can be used to obfuscate nefarious activities in log files, as well as to confuse network services that are highly dependent upon an accurate system time (such as sshd). All changes to the system time should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-001487, CCI-000169, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.4.2.b, 4.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Record Attempts to Alter the localtime File [ref] rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-w /etc/localtime -p wa -k audit_time_rules
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-w /etc/localtime -p wa -k audit_time_rules
```

The -k option allows for the specification of a key in string form that can be used for better reporting capability through ausearch and aureport and should always be used.

Rationale:
Arbitrary changes to the system time can be used to obfuscate nefarious activities in log files, as well as to confuse network services that are highly dependent upon an accurate system time (such as sshd). All changes to the system time should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-001487, CCI-000169, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.4.2.b, 4.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Record Events that Modify the System's Mandatory Access Controls [ref]

rule

If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d`:

```
-w /etc/selinux/ -p wa -k MAC-policy
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file:

```
-w /etc/selinux/ -p wa -k MAC-policy
```

Rationale:
The system's mandatory access policy (SELinux) should not be arbitrarily changed by anything other than administrator action. All changes to MAC policy should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.8, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.5.5, 4.1.6

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Ensure auditd Collects Information on Exporting to Media (successful) [ref]

rule

At a minimum, the audit system should collect media exportation events for all users and root. If the `auditd` daemon is configured to use the `augenrules` program to read audit rules during daemon startup (the default), add the following line to a file with suffix `.rules` in the directory `/etc/audit/rules.d`, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S mount -F auid>=1000 -F auid!=unset -F key=export
```

If the `auditd` daemon is configured to use the `auditctl` utility to read audit rules during daemon startup, add the following line to `/etc/audit/audit.rules` file, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S mount -F auid>=1000 -F auid!=unset -F key=export
```

Rationale:

The unauthorized exportation of data to external media could result in an information leak where classified information, Privacy Act information, and intellectual property could be lost. An audit trail should be created each time a filesystem is mounted to help identify and guard against information loss.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000135, CCI-000169, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.2.7, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SV-204552r603261_rule, 4.1.12

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Record Events that Modify the System's Network Environment [ref]

rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following lines to a file with suffix .rules in the directory /etc/audit/rules.d, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S sethostname,setdomainname -F key=audit_rules_networkconfig_modification
-w /etc/issue -p wa -k audit_rules_networkconfig_modification
-w /etc/issue.net -p wa -k audit_rules_networkconfig_modification
-w /etc/hosts -p wa -k audit_rules_networkconfig_modification
-w /etc/sysconfig/network -p wa -k audit_rules_networkconfig_modification
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following lines to /etc/audit/audit.rules file, setting ARCH to either b32 or b64 as appropriate for your system:

```
-a always,exit -F arch=ARCH -S sethostname,setdomainname -F key=audit_rules_networkconfig_modification
-w /etc/issue -p wa -k audit_rules_networkconfig_modification
-w /etc/issue.net -p wa -k audit_rules_networkconfig_modification
-w /etc/hosts -p wa -k audit_rules_networkconfig_modification
-w /etc/sysconfig/network -p wa -k audit_rules_networkconfig_modification
```

Rationale:
The network environment should not be modified by anything other than administrator action. Any change to network parameters should be audited.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.5.5, 4.1.5

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Ensure auditd Collects System Administrator Actions [ref]

rule

At a minimum, the audit system should collect administrator actions for all users and root. If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following line to a file with suffix .rules in the directory /etc/audit/rules.d :

```
-w /etc/sudoers -p wa -k actions
-w /etc/sudoers.d/ -p wa -k actions
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following line to /etc/audit/audit.rules file:

```
-w /etc/sudoers -p wa -k actions
-w /etc/sudoers.d/ -p wa -k actions
```

Rationale:
The actions taken by system administrators should be audited to keep a record of what was executed on the system, as well as, for accountability purposes.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 18, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, DSS06.03, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000126, CCI-000130, CCI-000135, CCI-000169, CCI-000172, CCI-002884, 164.308(a)(1)(ii)(D), 164.308(a)(3)(ii)(A), 164.308(a)(5)(ii)(C), 164.312(a)(2)(i), 164.312(b), 164.312(d), 164.312(e), 4.2.3.10, 4.3.2.6.7, 4.3.3.2.2, 4.3.3.3.9, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.8, 4.3.3.6.6, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.1, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.1.2, A.6.2.1, A.6.2.2, A.7.1.1, A.9.1.2, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.1, A.9.4.2, A.9.4.3, A.9.4.4, A.9.4.5, AC-2(7)(b), AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-1, PR.AC-3, PR.AC-4, PR.AC-6, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, FAU_GEN.1.1.c, Req-10.2.2, Req-10.2.5.b, SRG-OS-000004-GPOS-00004, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000062-GPOS-00031, SRG-OS-000304-GPOS-00121, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000470-GPOS-00214, SRG-OS-000471-GPOS-00215, SRG-OS-000239-GPOS-00089, SRG-OS-000240-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000304-GPOS-00121, SRG-OS-000466-GPOS-00210, SRG-OS-000476-GPOS-00221, SRG-OS-000462-VMM-001840, SRG-OS-000471-VMM-001910, SV-204549r603261_rule, 4.1.14

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Record Events that Modify User/Group Information [ref] rule

If the auditd daemon is configured to use the augenrules program to read audit rules during daemon startup (the default), add the following lines to a file with suffix .rules in the directory /etc/audit/rules.d , in order to capture events that modify account changes:

```
-w /etc/group -p wa -k audit_rules_usergroup_modification
-w /etc/passwd -p wa -k audit_rules_usergroup_modification
-w /etc/gshadow -p wa -k audit_rules_usergroup_modification
-w /etc/shadow -p wa -k audit_rules_usergroup_modification
-w /etc/security/opasswd -p wa -k audit_rules_usergroup_modification
```

If the auditd daemon is configured to use the auditctl utility to read audit rules during daemon startup, add the following lines to /etc/audit/audit.rules file, in order to capture events that modify account changes:

```
-w /etc/group -p wa -k audit_rules_usergroup_modification
-w /etc/passwd -p wa -k audit_rules_usergroup_modification
-w /etc/gshadow -p wa -k audit_rules_usergroup_modification
-w /etc/shadow -p wa -k audit_rules_usergroup_modification
-w /etc/security/opasswd -p wa -k audit_rules_usergroup_modification
```

Warning: This rule checks for multiple syscalls related to account changes; it was written with DISA STIG in mind. Other policies should use a separate rule for each syscall that needs to be checked. For example:

- audit_rules_usergroup_modification_group
- audit_rules_usergroup_modification_gshadow
- audit_rules_usergroup_modification_passwd

Rationale:

In addition to auditing new user and group accounts, these watches will alert the system administrator(s) to any modifications. Any unexpected users, groups, or modifications should be investigated for legitimacy.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 18, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, DSS06.03, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.1.7, CCI-000018, CCI-000130, CCI-000172, CCI-001403, CCI-002130, 4.2.3.10, 4.3.2.6.7, 4.3.3.2.2, 4.3.3.3.9, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.8, 4.3.3.6.6, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.1, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.1.2, A.6.2.1, A.6.2.2, A.7.1.1, A.9.1.2, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.1, A.9.4.2, A.9.4.3, A.9.4.4, A.9.4.5, CIP-004-6 R2.2.2, CIP-004-6 R2.2.3, CIP-007-3 R.1.3, CIP-007-3 R5, CIP-007-3 R5.1.1, CIP-007-3 R5.1.3, CIP-007-3 R5.2.1, CIP-007-3 R5.2.3, AC-2(4), AU-2(d), AU-12(c), AC-6(9), CM-6(a), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-1, PR.AC-3, PR.AC-4, PR.AC-6, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.2.5, SRG-OS-000004-GPOS-00004, SRG-OS-000037-GPOS-00015, SRG-OS-000042-GPOS-00020, SRG-OS-000239-GPOS-00089, SRG-OS-000241-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000392-GPOS-00172, SRG-OS-000462-GPOS-00206, SRG-OS-000471-GPOS-00215, SRG-OS-000476-GPOS-00221, SV-204550r603261_rule

Remediation Shell script: (show)

Ensure the audit Subsystem is Installed [ref]

rule

The audit package should be installed.

Rationale:
The auditd service is an access monitoring and accounting daemon, watching system calls to audit any access, in comparison with potential local access control policy such as SELinux policy.

Severity: medium

References: BP28(R50), CCI-000172, CCI-001814, CCI-001875, CCI-001877, CCI-001878, CCI-001879, CCI-001880, CCI-001881, CCI-001882, CCI-001889, CCI-001914, CCI-000169, CIP-004-6 R3.3, CIP-007-3 R6.5, AC-7(a), AU-7(1), AU-7(2), AU-14, AU-12(2), AU-2(a), CM-6(a), SRG-OS-000122-GPOS-00063, SRG-OS-000337-GPOS-00129, SRG-OS-000348-GPOS-00136, SRG-OS-000349-GPOS-00137, SRG-OS-000350-GPOS-00138, SRG-OS-000351-GPOS-00139, SRG-OS-000352-GPOS-00140, SRG-OS-000353-GPOS-00141, SRG-OS-000354-GPOS-00142, SRG-OS-000358-GPOS-00145, SRG-OS-000359-GPOS-00146, SRG-OS-000365-GPOS-00152, SRG-OS-000474-GPOS-00219, SRG-OS-000475-GPOS-00220, SRG-OS-000480-GPOS-00227, SRG-OS-000062-GPOS-00031, TAMU-AU-2(1), TAMU-AU-3(1.6), 4.1.1.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Remediation script: (show)

Enable auditd Service [ref]

rule

The auditd service is an essential userspace component of the Linux Auditing System, as it is responsible for writing audit records to disk. The auditd service can be enabled with the following command:

```
$ sudo systemctl enable auditd.service
```

Rationale:
Without establishing what type of events occurred, it would be difficult to establish, correlate, and investigate the events leading up to an outage or attack. Ensuring the auditd service is active ensures audit records generated by the kernel are appropriately recorded.

Additionally, a properly configured audit subsystem ensures that actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 19, 2, 3, 4, 5, 6, 7, 8, 9, 5.4.1.1, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO12.06, APO13.01, BAI03.05, BAI08.02, DSS01.03, DSS01.04, DSS02.02, DSS02.04, DSS02.07, DSS03.01, DSS03.05, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, 3.3.1, 3.3.2, 3.3.6, CCI-000126, CCI-000130, CCI-000131, CCI-000132, CCI-000133, CCI-000134, CCI-000135, CCI-000154, CCI-000158, CCI-000366, CCI-001464, CCI-001487, CCI-001814, CCI-001876, CCI-002884, CCI-000169, 164.308(a)(1)(ii)(D), 164.308(a)(5)(ii)(C), 164.310(a)(2)(iv), 164.310(d)(2)(iii), 164.312(b), 4.2.3.10, 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.3.6.6, 4.3.4.4.7, 4.3.4.5.6, 4.3.4.5.7, 4.3.4.5.8, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 1.13, SR 2.10, SR 2.11, SR 2.12, SR 2.6, SR 2.8, SR 2.9, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 6.1, SR 6.2, SR 7.1, SR 7.6, A.11.2.6, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.7, A.15.2.1, A.15.2.2, A.16.1.4, A.16.1.5, A.16.1.7, A.6.2.1, A.6.2.2, CIP-004-6 R3.3, CIP-007-3 R6.5, AC-2(g), AU-3, AU-10, AU-2(d), AU-12(c), AU-14(1), AC-6(9), CM-6(a), SI-4(23), DE.AE-3, DE.AE-5, DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.AC-3, PR.PT-1, PR.PT-4, RS.AN-1, RS.AN-4, Req-10.1, SRG-OS-000037-GPOS-00015, SRG-OS-000038-GPOS-00016, SRG-OS-000039-GPOS-00017, SRG-OS-000040-GPOS-00018, SRG-OS-000041-GPOS-00019, SRG-OS-000042-GPOS-00021, SRG-OS-000051-GPOS-00024, SRG-OS-000054-GPOS-00025, SRG-OS-000122-GPOS-00063, SRG-OS-000254-GPOS-00095, SRG-OS-000255-GPOS-00096, SRG-OS-000365-GPOS-00152, SRG-OS-000392-GPOS-00172, SRG-OS-000480-GPOS-00227, SRG-OS-000062-GPOS-00031, TAMU-AU-2(1), TAMU-AU-3(1.6), SRG-OS-000037-VMM-000150, SRG-OS-000063-VMM-000310, SRG-OS-000038-VMM-000160, SRG-OS-000039-VMM-000170, SRG-OS-000040-VMM-000180, SRG-OS-000041-VMM-000190, SV-204503r603261_rule, 4.1.1.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Configure Syslog [ref]

group

The syslog service has been the default Unix logging mechanism for many years. It has a number of downsides, including inconsistent log format, lack of authentication for received messages, and lack of authentication, encryption, or reliable transport for messages sent over a network. However, due to its long history, syslog is a de facto standard which is supported by almost all Unix applications.

In Red Hat Enterprise Linux 7, rsyslog has replaced ksyslogd as the syslog daemon of choice, and it includes some additional security features such as reliable, 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. This section discusses how to configure rsyslog for best effect, and how to use tools provided with the system to maintain and monitor logs.

▼ contains 6 rules

Ensure Proper Configuration of Log Files [ref]

group

The file /etc/rsyslog.conf controls where log message are written. These are controlled by lines called *rules*, which consist of a *selector* and an *action*. These rules are often customized depending on the role of the system, the requirements of the environment, and whatever may enable the administrator to most effectively make use of log data. The default rules in Red Hat Enterprise Linux 7 are:

```
*.info;mail.none;authpriv.none;cron.none                /var/log/messages
authpriv.*                                              /var/log/secure
mail.*                                                  -/var/log/maillog
cron.*                                                  /var/log/cron
*.emerg                                                 *
uucp,news.crit                                          /var/log/spooler
local7.*                                                /var/log/boot.log
```

See the man page rsyslog.conf(5) for more information. *Note that the rsyslog daemon can be configured to use a timestamp format that some log processing programs may not understand. If this occurs, edit the file /etc/rsyslog.conf and add or edit the following line:*

```
$ ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat
```

▼ contains 3 rules

Ensure Log Files Are Owned By Appropriate Group [ref]

rule

The group-owner of all log files written by rsyslog should be root. These log files are determined by the second part of each Rule line in /etc/rsyslog.conf and typically all appear in /var/log. For each log file *LOGFILE* referenced in /etc/rsyslog.conf, run the following command to inspect the file's group owner:

```
$ ls -l LOGFILE
```

If the owner is not root, run the following command to correct this:

```
$ sudo chgrp root LOGFILE
```

Rationale:

The log files generated by rsyslog contain valuable information regarding system configuration, user authentication, and other such information. Log files should be protected from unauthorized access.

Severity: medium

References: BP28(R46), BP28(R5), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, CCI-001314, 4.3.3.7.3, SR 2.1, SR 5.2, 0988, 1405, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-10.5.1, Req-10.5.2, TAMU-AU-2(1), TAMU-AU-9

Ensure Log Files Are Owned By Appropriate User [ref]

rule

The owner of all log files written by rsyslog should be root. These log files are determined by the second part of each Rule line in /etc/rsyslog.conf and typically all appear in /var/log. For each log file *LOGFILE* referenced in /etc/rsyslog.conf, run the following command to inspect the file's owner:

```
$ ls -l LOGFILE
```

If the owner is not root, run the following command to correct this:

```
$ sudo chown root LOGFILE
```

Rationale:
The log files generated by rsyslog contain valuable information regarding system configuration, user authentication, and other such information. Log files should be protected from unauthorized access.

Severity: medium

References: BP28(R46), BP28(R5), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, CCI-001314, 4.3.3.7.3, SR 2.1, SR 5.2, 0988, 1405, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-10.5.1, Req-10.5.2, TAMU-AU-2(1), TAMU-AU-9

Ensure System Log Files Have Correct Permissions [ref]

rule

The file permissions for all log files written by rsyslog should be set to 600, or more restrictive. These log files are determined by the second part of each Rule line in /etc/rsyslog.conf and typically all appear in /var/log. For each log file *LOGFILE* referenced in /etc/rsyslog.conf, run the following command to inspect the file's permissions:

```
$ ls -l LOGFILE
```

If the permissions are not 600 or more restrictive, run the following command to correct this:

```
$ sudo chmod 0600 LOGFILE
```

"

Rationale:
Log files can contain valuable information regarding system configuration. If the system log files are not protected unauthorized users could change the logged data, eliminating their forensic value.

Severity: medium

References: BP28(R36), CCI-001314, 0988, 1405, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), Req-10.5.1, Req-10.5.2, TAMU-AU-2(1), TAMU-AU-9, 4.2.1.3

Remediation Shell script: (show)

Ensure All Logs are Rotated by logrotate [ref]

group

Edit the file `/etc/logrotate.d/syslog`. Find the first line, which should look like this (wrapped for clarity):

```
/var/log/messages /var/log/secure /var/log/maillog /var/log/spooler \
  /var/log/boot.log /var/log/cron {
```

Edit this line so that it contains a one-space-separated listing of each log file referenced in `/etc/rsyslog.conf`.

All logs in use on a system must be rotated regularly, or the log files will consume disk space over time, eventually interfering with system operation. The file `/etc/logrotate.d/syslog` is the configuration file used by the `logrotate` program to maintain all log files written by `syslog`. By default, it rotates logs weekly and stores four archival copies of each log. These settings can be modified by editing `/etc/logrotate.conf`, but the defaults are sufficient for purposes of this guide.

Note that `logrotate` is run nightly by the cron job `/etc/cron.daily/logrotate`. If particularly active logs need to be rotated more often than once a day, some other mechanism must be used.

▼ contains 1 rule

Ensure Logrotate Runs Periodically [ref]

rule

The `logrotate` utility allows for the automatic rotation of log files. The frequency of rotation is specified in `/etc/logrotate.conf` which triggers a cron task. The value for the `rotate` paramater is 30. To configure logrotate to run daily, add or correct the following line in `/etc/logrotate.conf`:

```
# rotate log files frequency
daily
rotate 30
```

Rationale:
Log files that are not properly rotated run the risk of growing so large that they fill up the /var/log partition. Valuable logging information could be lost if the /var/log partition becomes full.

Severity: medium

References: BP28(R43), NT12(R18), 1, 14, 15, 16, 3, 5, 6, APO11.04, BAI03.05, DSS05.04, DSS05.07, MEA02.01, CCI-000366, 4.3.3.3.9, 4.3.3.5.8, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 2.10, SR 2.11, SR 2.12, SR 2.8, SR 2.9, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, CM-6(a), PR.PT-1, Req-10.7, TAMU-AU-11(1.2), 4.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Ensure rsyslog is Installed [ref]

rule

Rsyslog is installed by default. The rsyslog package can be installed with the following command:

```
$ sudo yum install rsyslog
```

Rationale:

The rsyslog package provides the rsyslog daemon, which provides system logging services.

Severity: medium

References: BP28(R5), NT28(R46), 1, 14, 15, 16, 3, 5, 6, APO11.04, BAI03.05, DSS05.04, DSS05.07, MEA02.01, CCI-001311, CCI-001312, CCI-000366, 164.312(a)(2)(ii), 4.3.3.3.9, 4.3.3.5.8, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 2.10, SR 2.11, SR 2.12, SR 2.8, SR 2.9, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, CM-6(a), PR.PT-1, SRG-OS-000479-GPOS-00224, SRG-OS-000051-GPOS-00024, SRG-OS-000480-GPOS-00227, TAMU-AU-2(1), TAMU-AU-3, 4.2.1.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Remediation script: (show)

Enable rsyslog Service [ref]

rule

The rsyslog service provides syslog-style logging by default on Red Hat Enterprise Linux 7. The rsyslog service can be enabled with the following command:

```
$ sudo systemctl enable rsyslog.service
```

Rationale:

The rsyslog service must be running in order to provide logging services, which are essential to system administration.

Severity: medium

References: BP28(R5), NT28(R46), 1, 12, 13, 14, 15, 16, 2, 3, 5, 6, 7, 8, 9, APO10.01, APO10.03, APO10.04, APO10.05, APO11.04, APO13.01, BAI03.05, BAI04.04, DSS01.03, DSS03.05, DSS05.02, DSS05.04, DSS05.05, DSS05.07, MEA01.01, MEA01.02, MEA01.03, MEA01.04, MEA01.05, MEA02.01, CCI-001311, CCI-001312, CCI-001557, CCI-001851, CCI-000366, 164.312(a)(2)(ii), 4.3.2.6.7, 4.3.3.3.9, 4.3.3.5.8, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4, SR 2.10, SR 2.11, SR 2.12, SR 2.8, SR 2.9, SR 6.1, SR 6.2, SR 7.1, SR 7.2, A.12.1.3, A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1, A.14.2.7, A.15.2.1, A.15.2.2, A.17.2.1, CM-6(a), AU-4(1), DE.CM-1, DE.CM-3, DE.CM-7, ID.SC-4, PR.DS-4, PR.PT-1, SRG-OS-000480-GPOS-00227, TAMU-AU-2(1), TAMU-AU-3, 4.2.1.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

File Permissions and Masks [ref]

group

Traditional Unix security relies heavily on file and directory permissions to prevent unauthorized users from reading or modifying files to which they should not have access.

Several of the commands in this section search filesystems for files or directories with certain characteristics, and are intended to be run on every local partition on a given system. When the variable *PART* appears in one of the commands below, it means that the command is intended to be run repeatedly, with the name of each local partition substituted for *PART* in turn.

The following command prints a list of all xfs partitions on the local system, which is the default filesystem for Red Hat Enterprise Linux 7 installations:

```
$ mount -t xfs | awk '{print $3}'
```

For any systems that use a different local filesystem type, modify this command as appropriate.

▼ contains 46 rules

Verify Permissions on Important Files and Directories [ref]

group

Permissions for many files on a system must be set restrictively to ensure sensitive information is properly protected. This section discusses important permission restrictions which can be verified to ensure that no harmful discrepancies have arisen.

▼ contains 31 rules

Verify Permissions on Files with Local Account Information and Credentials [ref]

group

The default restrictive permissions for files which act as important security databases such as `passwd`, `shadow`, `group`, and `gshadow` files must be maintained. Many utilities need read access to the `passwd` file in order to function properly, but read access to the `shadow` file allows malicious attacks against system passwords, and should never be enabled.

▼ contains 24 rules

Verify Group Who Owns Backup group File [ref]

rule

To properly set the group owner of `/etc/group-`, run the command:

```
$ sudo chgrp root /etc/group-
```

Rationale:

The `/etc/group-` file is a backup file of `/etc/group`, and as such, it contains information regarding groups that are configured on the system. Protection of this file is important for system security.

Severity: medium

References: 6.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns Backup gshadow File [ref]

rule

To properly set the group owner of /etc/gshadow- , run the command:

```
$ sudo chgrp root /etc/gshadow-
```

Rationale:
The /etc/gshadow- file is a backup of /etc/gshadow , and as such, it contains group password hashes. Protection of this file is critical for system security.

Severity: medium

References: 6.1.6

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns Backup passwd File [ref]

rule

To properly set the group owner of /etc/passwd- , run the command:

```
$ sudo chgrp root /etc/passwd-
```

Rationale:
The /etc/passwd- file is a backup file of /etc/passwd , and as such, it contains information about the users that are configured on the system. Protection of this file is critical for system security.

Severity: medium

References: 6.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns Backup shadow File [ref]

rule

To properly set the group owner of /etc/shadow- , run the command:

```
$ sudo chgrp root /etc/shadow-
```

Rationale:
The /etc/shadow- file is a backup file of /etc/shadow , and as such, it contains the list of local system accounts and password hashes. Protection of this file is critical for system security.

Severity: medium

References: 6.1.5

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns group File [ref]

rule

To properly set the group owner of /etc/group , run the command:

```
$ sudo chgrp root /etc/group
```

Rationale:
The /etc/group file contains information regarding groups that are configured on the system. Protection of this file is important for system security.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns gshadow File [ref]

rule

To properly set the group owner of /etc/gshadow , run the command:

```
$ sudo chgrp root /etc/gshadow
```

Rationale:
The /etc/gshadow file contains group password hashes. Protection of this file is critical for system security.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, 6.1.7

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns passwd File [ref] rule

To properly set the group owner of /etc/passwd, run the command:

```
$ sudo chgrp root /etc/passwd
```

Rationale:
The /etc/passwd file contains information about the users that are configured on the system. Protection of this file is critical for system security.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns shadow File [ref] rule

To properly set the group owner of /etc/shadow, run the command:

```
$ sudo chgrp root /etc/shadow
```

Rationale:
The /etc/shadow file stores password hashes. Protection of this file is critical for system security.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns Backup group File [ref] rule

To properly set the owner of /etc/group-, run the command:

```
$ sudo chown root /etc/group-
```

Rationale:
The /etc/group- file is a backup file of /etc/group, and as such, it contains information regarding groups that are configured on the system. Protection of this file is important for system security.

Severity: medium

References: 6.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns Backup gshadow File [ref]

rule

To properly set the owner of /etc/gshadow- , run the command:

```
$ sudo chown root /etc/gshadow-
```

Rationale:
The /etc/gshadow- file is a backup of /etc/gshadow , and as such, it contains group password hashes. Protection of this file is critical for system security.

Severity: medium

References: 6.1.6

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns Backup passwd File [ref]

rule

To properly set the owner of /etc/passwd- , run the command:

```
$ sudo chown root /etc/passwd-
```

Rationale:
The /etc/passwd- file is a backup file of /etc/passwd , and as such, it contains information about the users that are configured on the system. Protection of this file is critical for system security.

Severity: medium

References: 6.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Group Who Owns Backup shadow File [ref]

rule

To properly set the owner of /etc/shadow- , run the command:

```
$ sudo chown root /etc/shadow-
```

Rationale:
The /etc/shadow- file is a backup file of /etc/shadow , and as such, it contains the list of local system accounts and password hashes. Protection of this file is critical for system security.

Severity: medium

References: 6.1.5

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns group File [ref]

rule

To properly set the owner of /etc/group , run the command:

```
$ sudo chown root /etc/group
```

Rationale:
The /etc/group file contains information regarding groups that are configured on the system. Protection of this file is important for system security.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns gshadow File [ref]

rule

To properly set the owner of /etc/gshadow , run the command:

```
$ sudo chown root /etc/gshadow
```

Rationale:
The /etc/gshadow file contains group password hashes. Protection of this file is critical for system security.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, 6.1.7

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns passwd File [ref]

rule

To properly set the owner of /etc/passwd, run the command:

```
$ sudo chown root /etc/passwd
```

Rationale:
The /etc/passwd file contains information about the users that are configured on the system. Protection of this file is critical for system security.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify User Who Owns shadow File [ref]

rule

To properly set the owner of /etc/shadow, run the command:

```
$ sudo chown root /etc/shadow
```

Rationale:
The /etc/shadow file contains the list of local system accounts and stores password hashes. Protection of this file is critical for system security. Failure to give ownership of this file to root provides the designated owner with access to sensitive information which could weaken the system security posture.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on Backup group File [ref]

rule

To properly set the permissions of /etc/group- , run the command:

```
$ sudo chmod 0644 /etc/group-
```

Rationale:
The /etc/group- file is a backup file of /etc/group , and as such, it contains information regarding groups that are configured on the system. Protection of this file is important for system security.

Severity: medium

References: 6.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on Backup gshadow File [ref]

rule

To properly set the permissions of /etc/gshadow- , run the command:

```
$ sudo chmod 0000 /etc/gshadow-
```

Rationale:
The /etc/gshadow- file is a backup of /etc/gshadow , and as such, it contains group password hashes. Protection of this file is critical for system security.

Severity: medium

References: 6.1.6

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on Backup passwd File [ref]

rule

To properly set the permissions of /etc/passwd- , run the command:

```
$ sudo chmod 0644 /etc/passwd-
```

Rationale:
The /etc/passwd- file is a backup file of /etc/passwd , and as such, it contains information about the users that are configured on the system. Protection of this file is critical for system security.

Severity: medium

References: 6.1.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on Backup shadow File [ref]

rule

To properly set the permissions of /etc/shadow-, run the command:

```
$ sudo chmod 0000 /etc/shadow-
```

Rationale:
The /etc/shadow- file is a backup file of /etc/shadow, and as such, it contains the list of local system accounts and password hashes. Protection of this file is critical for system security.

Severity: medium

References: 6.1.5

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on group File [ref]

rule

To properly set the permissions of /etc/passwd, run the command:

```
$ sudo chmod 0644 /etc/passwd
```

Rationale:
The /etc/group file contains information regarding groups that are configured on the system. Protection of this file is important for system security.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on gshadow File [ref]

rule

To properly set the permissions of /etc/gshadow , run the command:

```
$ sudo chmod 0000 /etc/gshadow
```

Rationale:
The /etc/gshadow file contains group password hashes. Protection of this file is critical for system security.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, 6.1.7

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on passwd File [ref]

rule

To properly set the permissions of /etc/passwd , run the command:

```
$ sudo chmod 0644 /etc/passwd
```

Rationale:
If the /etc/passwd file is writable by a group-owner or the world the risk of its compromise is increased. The file contains the list of accounts on the system and associated information, and protection of this file is critical for system security.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify Permissions on shadow File [ref]

rule

To properly set the permissions of /etc/shadow, run the command:

```
$ sudo chmod 0000 /etc/shadow
```

Rationale:
The /etc/shadow file contains the list of local system accounts and stores password hashes. Protection of this file is critical for system security. Failure to give ownership of this file to root provides the designated owner with access to sensitive information which could weaken the system security posture.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, 5.5.2.2, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, Req-8.7.c, 6.1.4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify that All World-Writable Directories Have Sticky Bits Set [ref]

rule

When the so-called 'sticky bit' is set on a directory, only the owner of a given file may remove that file from the directory. Without the sticky bit, any user with write access to a directory may remove any file in the directory. Setting the sticky bit prevents users from removing each other's files. In cases where there is no reason for a directory to be world-writable, a better solution is to remove that permission rather than to set the sticky bit. However, if a directory is used by a particular application, consult that application's documentation instead of blindly changing modes.
To set the sticky bit on a world-writable directory *DIR*, run the following command:

```
$ sudo chmod +t DIR
```

Rationale:
Failing to set the sticky bit on public directories allows unauthorized users to delete files in the directory structure.

The only authorized public directories are those temporary directories supplied with the system, or those designed to be temporary file repositories. The setting is normally reserved for directories used by the system, by users for temporary file storage (such as /tmp), and for directories requiring global read/write access.

Severity: medium

References: BP28(R40), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, CCI-001090, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, SRG-OS-000138-GPOS-00069, 1.1.22

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Verify that local System.map file (if exists) is readable only by root [ref] rule

Files containing sensitive informations should be protected by restrictive permissions. Most of the time, there is no need that these files need to be read by any non-root user To properly set the permissions of `/boot/System.map-*`, run the command:

```
$ sudo chmod 0600 /boot/System.map-*
```

Rationale:
The `System.map` file contains information about kernel symbols and can give some hints to generate local exploitation.

Severity: unknown

References: BP28(R13)

Ensure All SGID Executables Are Authorized [ref] rule

The SGID (set group id) bit should be set only on files that were installed via authorized means. A straightforward means of identifying unauthorized SGID files is determine if any were not installed as part of an RPM package, which is cryptographically verified. Investigate the origin of any unpackaged SGID files. This configuration check considers authorized SGID files which were installed via RPM. It is assumed that when an individual has sudo access to install an RPM and all packages are signed with an organizationally-recognized GPG key, the software should be considered an approved package on the system. Any SGID file not deployed through an RPM will be flagged for further review.

Rationale:
Executable files with the SGID permission run with the privileges of the owner of the file. SGID files of uncertain provenance could allow for unprivileged users to elevate privileges. The presence of these files should be strictly controlled on the system.

Severity: medium

References: BP28(R37), BP28(R38), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, 6.1.14

Ensure All SUID Executables Are Authorized [ref] rule

The SUID (set user id) bit should be set only on files that were installed via authorized means. A straightforward means of identifying unauthorized SUID files is determine if any were not installed as part of an RPM package, which is cryptographically verified. Investigate the origin of any unpackaged SUID files. This configuration check considers authorized SUID files which were installed via RPM. It is assumed that when an individual has sudo access to install an RPM and all packages are signed with an organizationally-recognized GPG key, the software should be considered an approved package on the system. Any SUID file not deployed through an RPM will be flagged for further review.

Rationale:
Executable files with the SUID permission run with the privileges of the owner of the file. SUID files of uncertain provenance could allow for unprivileged users to elevate privileges. The presence of these files should be strictly controlled on the system.

Severity: medium

References: BP28(R37), BP28(R38), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, 6.1.13

Ensure No World-Writable Files Exist [ref]

rule

It is generally a good idea to remove global (other) write access to a file when it is discovered. However, check with documentation for specific applications before making changes. Also, monitor for recurring world-writable files, as these may be symptoms of a misconfigured application or user account. Finally, this applies to real files and not virtual files that are a part of pseudo file systems such as `sysfs` or `procfs` .

Rationale:
Data in world-writable files can be modified by any user on the system. In almost all circumstances, files can be configured using a combination of user and group permissions to support whatever legitimate access is needed without the risk caused by world-writable files.

Severity: medium

References: BP28(R40), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, 6.1.10

Remediation Shell script: (show)

Enable Kernel Parameter to Enforce DAC on Hardlinks [ref]

rule

To set the runtime status of the `fs.protected_hardlinks` kernel parameter, run the following command:

```
$ sudo sysctl -w fs.protected_hardlinks=1
```

To make sure that the setting is persistent, add the following line to a file in the directory `/etc/sysctl.d` :

```
fs.protected_hardlinks = 1
```

Rationale:
By enabling this kernel parameter, users can no longer create soft or hard links to files which they do not own. Disallowing such hardlinks mitigate vulnerabilities based on insecure file system accessed by privileged programs, avoiding an exploitation vector exploiting unsafe use of `open()` or `creat()` .

Severity: medium

References: BP28(R23), CCI-002165, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), SRG-OS-000324-GPOS-00125

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Enable Kernel Parameter to Enforce DAC on Symlinks [ref]

rule

To set the runtime status of the fs.protected_symlinks kernel parameter, run the following command:

```
$ sudo sysctl -w fs.protected_symlinks=1
```

To make sure that the setting is persistent, add the following line to a file in the directory /etc/sysctl.d :

```
fs.protected_symlinks = 1
```

Rationale:

By enabling this kernel parameter, symbolic links are permitted to be followed only when outside a sticky world-writable directory, or when the UID of the link and follower match, or when the directory owner matches the symlink's owner. Disallowing such symlinks helps mitigate vulnerabilities based on insecure file system accessed by privileged programs, avoiding an exploitation vector exploiting unsafe use of open() or creat() .

Severity: medium

References: BP28(R23), CCI-002165, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-6(1), SRG-OS-000324-GPOS-00125

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Restrict Dynamic Mounting and Unmounting of Filesystems [ref]

group

Linux includes a number of facilities for the automated addition and removal of filesystems on a running system. These facilities may be necessary in many environments, but this capability also carries some risk -- whether direct risk from allowing users to introduce arbitrary filesystems, or risk that software flaws in the automated mount facility itself could allow an attacker to compromise the system.

This command can be used to list the types of filesystems that are available to the currently executing kernel:

```
$ find /lib/modules/`uname -r`/kernel/fs -type f -name '*.ko'
```

If these filesystems are not required then they can be explicitly disabled in a configuratio file in /etc/modprobe.d .

▼ contains 1 rule

Disable the Automounter [ref]

rule

The autofs daemon mounts and unmounts filesystems, such as user home directories shared via NFS, on demand. In addition, autofs can be used to handle removable media, and the default configuration provides the cdrom device as /misc/cd. However, this method of providing access to removable media is not common, so autofs can almost always be disabled if NFS is not in use. Even if NFS is required, it may be possible to configure filesystem mounts statically by editing /etc/fstab rather than relying on the automounter.

The autofs service can be disabled with the following command:

```
$ sudo systemctl mask --now autofs.service
```

Rationale:
Disabling the automounter permits the administrator to statically control filesystem mounting through /etc/fstab.

Additionally, automatically mounting filesystems permits easy introduction of unknown devices, thereby facilitating malicious activity.

Severity: medium

References: 1, 12, 15, 16, 5, APO13.01, DSS01.04, DSS05.03, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.4.6, CCI-000366, CCI-000778, CCI-001958, 164.308(a)(3)(i), 164.308(a)(3)(ii)(A), 164.310(d)(1), 164.310(d)(2), 164.312(a)(1), 164.312(a)(2)(iv), 164.312(b), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.6, A.11.2.6, A.13.1.1, A.13.2.1, A.18.1.4, A.6.2.1, A.6.2.2, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, CM-7(a), CM-7(b), CM-6(a), MP-7, PR.AC-1, PR.AC-3, PR.AC-6, PR.AC-7, SRG-OS-000114-GPOS-00059, SRG-OS-000378-GPOS-00163, SRG-OS-000480-GPOS-00227, SV-204451r603261_rule, 1.1.23

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Restrict Partition Mount Options [ref]

group

System partitions can be mounted with certain options that limit what files on those partitions can do. These options are set in the /etc/fstab configuration file, and can be used to make certain types of malicious behavior more difficult.

▼ contains 12 rules

Add nodev Option to /dev/shm [ref]

rule

The nodev mount option can be used to prevent creation of device files in /dev/shm. Legitimate character and block devices should not exist within temporary directories like /dev/shm. Add the nodev option to the fourth column of /etc/fstab for the line which controls mounting of /dev/shm.

Rationale:
The only legitimate location for device files is the /dev directory located on the root partition. The only exception to this is chroot jails.

Severity: low

References: 11, 13, 14, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS05.06, DSS06.06, CCI-001764, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000368-GPOS-00154, 1.1.8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Add noexec Option to /dev/shm [ref]

rule

The noexec mount option can be used to prevent binaries from being executed out of /dev/shm. It can be dangerous to allow the execution of binaries from world-writable temporary storage directories such as /dev/shm. Add the noexec option to the fourth column of /etc/fstab for the line which controls mounting of /dev/shm.

Rationale:
Allowing users to execute binaries from world-writable directories such as /dev/shm can expose the system to potential compromise.

Severity: low

References: 11, 13, 14, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS05.06, DSS06.06, CCI-001764, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000368-GPOS-00154, SV-204486r603261_rule, 1.1.7

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Add nosuid Option to /dev/shm [ref]

rule

The nosuid mount option can be used to prevent execution of setuid programs in /dev/shm . The SUID and SGID permissions should not be required in these world-writable directories. Add the nosuid option to the fourth column of /etc/fstab for the line which controls mounting of /dev/shm .

Rationale:
The presence of SUID and SGID executables should be tightly controlled. Users should not be able to execute SUID or SGID binaries from temporary storage partitions.

Severity: low

References: 11, 13, 14, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS05.06, DSS06.06, CCI-001764, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000368-GPOS-00154, 1.1.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Add nodev Option to /home [ref]

rule

The nodev mount option can be used to prevent device files from being created in /home . Legitimate character and block devices should exist only in the /dev directory on the root partition or within chroot jails built for system services. Add the nodev option to the fourth column of /etc/fstab for the line which controls mounting of /home .

Rationale:
The only legitimate location for device files is the /dev directory located on the root partition. The only exception to this is chroot jails.

Severity: unknown

References: BP28(R12), SRG-OS-000368-GPOS-00154, 1.1.18

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Anaconda snippet: (show)

Add nodev Option to Removable Media Partitions [ref]

rule

The nodev mount option prevents files from being interpreted as character or block devices. Legitimate character and block devices should exist only in the /dev directory on the root partition or within chroot jails built for system services. Add the nodev option to the fourth column of /etc/fstab for the line which controls mounting of any removable media partitions.

Rationale:
The only legitimate location for device files is the /dev directory located on the root partition. An exception to this is chroot jails, and it is not advised to set nodev on partitions which contain their root filesystems.

Severity: low

References: 11, 12, 13, 14, 16, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.06, DSS05.07, DSS06.03, DSS06.06, CCI-000366, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.7.1.1, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, A.9.2.1, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.AC-3, PR.AC-6, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000480-GPOS-00227, 1.1.20

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Add noexec Option to Removable Media Partitions [ref]

rule

The noexec mount option prevents the direct execution of binaries on the mounted filesystem. Preventing the direct execution of binaries from removable media (such as a USB key) provides a defense against malicious software that may be present on such untrusted media. Add the noexec option to the fourth column of /etc/fstab for the line which controls mounting of any removable media partitions.

Rationale:
Allowing users to execute binaries from removable media such as USB keys exposes the system to potential compromise.

Severity: medium

References: 11, 12, 13, 14, 16, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.06, DSS05.07, DSS06.03, DSS06.06, CCI-000087, CCI-000366, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.7.1.1, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, A.9.2.1, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.AC-3, PR.AC-6, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000480-GPOS-00227, 1.1.19

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Add nosuid Option to Removable Media Partitions [ref]

rule

The nosuid mount option prevents set-user-identifier (SUID) and set-group-identifier (SGID) permissions from taking effect. These permissions allow users to execute binaries with the same permissions as the owner and group of the file respectively. Users should not be allowed to introduce SUID and SGID files into the system via partitions mounted from removeable media. Add the nosuid option to the fourth column of /etc/fstab for the line which controls mounting of any removable media partitions.

Rationale:
The presence of SUID and SGID executables should be tightly controlled. Allowing users to introduce SUID or SGID binaries from partitions mounted off of removable media would allow them to introduce their own highly-privileged programs.

Severity: medium

References: 11, 12, 13, 14, 15, 16, 18, 3, 5, 8, 9, APO01.06, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.06, DSS05.07, DSS06.02, DSS06.03, DSS06.06, CCI-000366, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 5.2, SR 7.6, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.11.2.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.1.2, A.6.2.1, A.6.2.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.1, A.9.1.2, A.9.2.1, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.AC-3, PR.AC-4, PR.AC-6, PR.DS-5, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000480-GPOS-00227, SV-204481r603261_rule, 1.1.21

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Add nodev Option to /tmp [ref]

rule

The nodev mount option can be used to prevent device files from being created in /tmp. Legitimate character and block devices should not exist within temporary directories like /tmp. Add the nodev option to the fourth column of /etc/fstab for the line which controls mounting of /tmp.

Rationale:
The only legitimate location for device files is the /dev directory located on the root partition. The only exception to this is chroot jails.

Severity: medium

References: BP28(R12), 11, 13, 14, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS05.06, DSS06.06, CCI-001764, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000368-GPOS-00154, 1.1.4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Anaconda snippet: (show)

Add nosuid Option to /tmp [ref]

rule

The nosuid mount option can be used to prevent execution of setuid programs in /tmp . The SUID and SGID permissions should not be required in these world-writable directories. Add the nosuid option to the fourth column of /etc/fstab for the line which controls mounting of /tmp .

Rationale:
The presence of SUID and SGID executables should be tightly controlled. Users should not be able to execute SUID or SGID binaries from temporary storage partitions.

Severity: medium

References: BP28(R12), 11, 13, 14, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS05.06, DSS06.06, CCI-001764, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.11.2.9, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.8.2.1, A.8.2.2, A.8.2.3, A.8.3.1, A.8.3.3, A.9.1.2, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-7(a), CM-7(b), CM-6(a), AC-6, AC-6(1), MP-7, PR.IP-1, PR.PT-2, PR.PT-3, SRG-OS-000368-GPOS-00154, 1.1.5

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Anaconda snippet: (show)

Add nodev Option to /var/tmp [ref]

rule

The nodev mount option can be used to prevent device files from being created in /var/tmp . Legitimate character and block devices should not exist within temporary directories like /var/tmp . Add the nodev option to the fourth column of /etc/fstab for the line which controls mounting of /var/tmp .

Rationale:
The only legitimate location for device files is the /dev directory located on the root partition. The only exception to this is chroot jails.

Severity: medium

References: BP28(R12), CCI-001764, SRG-OS-000368-GPOS-00154, 1.1.13

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Anaconda snippet: (show)

Add noexec Option to /var/tmp [ref]

rule

The noexec mount option can be used to prevent binaries from being executed out of /var/tmp . Add the noexec option to the fourth column of /etc/fstab for the line which controls mounting of /var/tmp .

Rationale:
Allowing users to execute binaries from world-writable directories such as /var/tmp should never be necessary in normal operation and can expose the system to potential compromise.

Severity: medium

References: BP28(R12), CCI-001764, SRG-OS-000368-GPOS-00154, 1.1.12

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Anaconda snippet: (show)

Add nosuid Option to /var/tmp [ref]

rule

The nosuid mount option can be used to prevent execution of setuid programs in /var/tmp . The SUID and SGID permissions should not be required in these world-writable directories. Add the nosuid option to the fourth column of /etc/fstab for the line which controls mounting of /var/tmp .

Rationale:
The presence of SUID and SGID executables should be tightly controlled. Users should not be able to execute SUID or SGID binaries from temporary storage partitions.

Severity: medium

References: BP28(R12), CCI-001764, SRG-OS-000368-GPOS-00154, 1.1.14

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Anaconda snippet: (show)

Restrict Programs from Dangerous Execution Patterns [ref]

group

The recommendations in this section are designed to ensure that the system's features to protect against potentially dangerous program execution are activated. These protections are applied at the system initialization or kernel level, and defend against certain types of badly-configured or compromised programs.

▼ contains 2 rules

Disable Core Dumps [ref]

group

A core dump file is the memory image of an executable program when it was terminated by the operating system due to errant behavior. In most cases, only software developers legitimately need to access these files. The core dump files may also contain sensitive information, or unnecessarily occupy large amounts of disk space.

Once a hard limit is set in `/etc/security/limits.conf`, or to a file within the `/etc/security/limits.d/` directory, a user cannot increase that limit within his or her own session. If access to core dumps is required, consider restricting them to only certain users or groups. See the `limits.conf` man page for more information.

The core dumps of setuid programs are further protected. The `sysctl` variable `fs.suid_dumpable` controls whether the kernel allows core dumps from these programs at all. The default value of 0 is recommended.

▼ contains 1 rule

Disable Core Dumps for SUID programs [ref]

rule

To set the runtime status of the `fs.suid_dumpable` kernel parameter, run the following command:

```
$ sudo sysctl -w fs.suid_dumpable=0
```

To make sure that the setting is persistent, add the following line to a file in the directory `/etc/sysctl.d`:

```
fs.suid_dumpable = 0
```

Rationale:

The core dump of a setuid program is more likely to contain sensitive data, as the program itself runs with greater privileges than the user who initiated execution of the program. Disabling the ability for any setuid program to write a core file decreases the risk of unauthorized access of such data.

Severity: medium

References: BP28(R23), 164.308(a)(1)(ii)(D), 164.308(a)(3), 164.308(a)(4), 164.310(b), 164.310(c), 164.312(a), 164.312(e), SI-11(a), SI-11(b), 1.5.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Enable ExecShield [ref]

group

ExecShield describes kernel features that provide protection against exploitation of memory corruption errors such as buffer overflows. These features include random placement of the stack and other memory regions, prevention of execution in memory that should only hold data, and special handling of text buffers. These protections are enabled by default on 32-bit systems and controlled through `sysctl` variables `kernel.exec-shield` and `kernel.randomize_va_space`. On the latest 64-bit systems, `kernel.exec-shield` cannot be enabled or disabled with `sysctl`.

▼ contains 1 rule

Enable Randomized Layout of Virtual Address Space [ref]

rule

To set the runtime status of the kernel.randomize_va_space kernel parameter, run the following command:

```
$ sudo sysctl -w kernel.randomize_va_space=2
```

To make sure that the setting is persistent, add the following line to a file in the directory /etc/sysctl.d :

```
kernel.randomize_va_space = 2
```

Rationale:

Address space layout randomization (ASLR) makes it more difficult for an attacker to predict the location of attack code they have introduced into a process's address space during an attempt at exploitation. Additionally, ASLR makes it more difficult for an attacker to know the location of existing code in order to re-purpose it using return oriented programming (ROP) techniques.

Severity: medium

References: BP28(R23), 3.1.7, CCI-000366, CCI-002824, 164.308(a)(1)(ii)(D), 164.308(a)(3), 164.308(a)(4), 164.310(b), 164.310(c), 164.312(a), 164.312(e), CIP-002-5 R1.1, CIP-002-5 R1.2, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 4.1, CIP-004-6 4.2, CIP-004-6 R2.2.3, CIP-004-6 R2.2.4, CIP-004-6 R2.3, CIP-004-6 R4, CIP-005-6 R1, CIP-005-6 R1.1, CIP-005-6 R1.2, CIP-007-3 R3, CIP-007-3 R3.1, CIP-007-3 R5.1, CIP-007-3 R5.1.2, CIP-007-3 R5.1.3, CIP-007-3 R5.2.1, CIP-007-3 R5.2.3, CIP-007-3 R8.4, CIP-009-6 R.1.1, CIP-009-6 R4, SC-30, SC-30(2), CM-6(a), SRG-OS-000433-GPOS-00193, SRG-OS-000480-GPOS-00227, SV-204584r603261_rule, 1.5.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Services [ref]

group

The best protection against vulnerable software is running less software. This section describes how to review the software which Red Hat Enterprise Linux 7 installs on a system and disable software which is not needed. It then enumerates the software packages installed on a default Red Hat Enterprise Linux 7 system and provides guidance about which ones can be safely disabled.

Red Hat Enterprise Linux 7 provides a convenient minimal install option that essentially installs the bare necessities for a functional system. When building Red Hat Enterprise Linux 7 systems, it is highly recommended to select the minimal packages and then build up the system from there.

▼ contains 31 rules

Base Services [ref]

group

This section addresses the base services that are installed on a Red Hat Enterprise Linux 7 default installation which are not covered in other sections. Some of these services listen on the network and should be treated with particular discretion. Other services are local system utilities that may or may not be extraneous. In general, system services should be disabled if not required.

▼ contains 4 rules

Disable Automatic Bug Reporting Tool (abrtd) [ref] rule

The Automatic Bug Reporting Tool (`abrtd`) daemon collects and reports crash data when an application crash is detected. Using a variety of plugins, abrtd can email crash reports to system administrators, log crash reports to files, or forward crash reports to a centralized issue tracking system such as RHTSupport. The `abrtd` service can be disabled with the following command:

```
$ sudo systemctl mask --now abrtd.service
```

Rationale:
Mishandling crash data could expose sensitive information about vulnerabilities in software executing on the system, as well as sensitive information from within a process's address space or registers.

Severity: medium

References: 11, 12, 14, 15, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.05, DSS06.06, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.11.2.6, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.9.1.2, CM-7(a), CM-6(a), PR.AC-3, PR.IP-1, PR.PT-3, PR.PT-4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Disable Odd Job Daemon (oddjobd) [ref]

rule

The oddjobd service exists to provide an interface and access control mechanism through which specified privileged tasks can run tasks for unprivileged client applications. Communication with oddjobd through the system message bus. The oddjobd service can be disabled with the following command:

```
$ sudo systemctl mask --now oddjobd.service
```

Rationale:

The oddjobd service may provide necessary functionality in some environments, and can be disabled if it is not needed. Execution of tasks by privileged programs, on behalf of unprivileged ones, has traditionally been a source of privilege escalation security issues.

Severity: medium

References: 11, 14, 3, 9, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS06.06, CCI-000381, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.9.1.2, CM-7(a), CM-7(b), CM-6(a), PR.IP-1, PR.PT-3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Disable Apache Qpid (qpidd) [ref]

rule

The qpidd service provides high speed, secure, guaranteed delivery services. It is an implementation of the Advanced Message Queuing Protocol. By default the qpidd service will bind to port 5672 and listen for connection attempts. The qpidd service can be disabled with the following command:

```
$ sudo systemctl mask --now qpidd.service
```

Rationale:
The qpidd service is automatically installed when the base package selection is selected during installation. The qpidd service listens for network connections, which increases the attack surface of the system. If the system is not intended to receive AMQP traffic, then the qpidd service is not needed and should be disabled or removed.

Severity: low

References: 11, 12, 14, 15, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.05, DSS06.06, CCI-000382, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.11.2.6, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.9.1.2, CM-7(a), CM-7(b), CM-6(a), PR.AC-3, PR.IP-1, PR.PT-3, PR.PT-4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Disable Network Router Discovery Daemon (rdisc) [ref]

rule

The rdisc service implements the client side of the ICMP Internet Router Discovery Protocol (IRDP), which allows discovery of routers on the local subnet. If a router is discovered then the local routing table is updated with a corresponding default route. By default this daemon is disabled. The rdisc service can be disabled with the following command:

```
$ sudo systemctl mask --now rdisc.service
```

Rationale:
General-purpose systems typically have their network and routing information configured statically by a system administrator. Workstations or some special-purpose systems often use DHCP (instead of IRDP) to retrieve dynamic network configuration information.

Severity: medium

References: 1, 11, 12, 13, 14, 15, 16, 18, 3, 4, 6, 8, 9, APO01.06, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS01.05, DSS03.01, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, DSS06.02, DSS06.06, CCI-000382, 4.2.3.4, 4.3.3.4, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, 4.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.11.2.6, A.12.1.1, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.1.2, A.13.1.3, A.13.2.1, A.13.2.2, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.1.2, A.6.2.1, A.6.2.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, AC-4, CM-7(a), CM-7(b), CM-6(a), DE.AE-1, ID.AM-3, PR.AC-3, PR.AC-5, PR.DS-5, PR.IP-1, PR.PT-3, PR.PT-4

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Cron and At Daemons [ref]

group

The cron and at services are used to allow commands to be executed at a later time. The cron service is required by almost all systems to perform necessary maintenance tasks, while at may or may not be required on a given system. Both daemons should be configured defensively.

▼ contains 3 rules

Install the cron service [ref]

rule

The Cron service should be installed.

Rationale:
The cron service allow periodic job execution, needed for almost all administrative tasks and services (software update, log rotating, etc.). Access to cron service should be restricted to administrative accounts only.

Severity: medium

References: BP28(R50), 11, 14, 3, 9, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS06.06, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.9.1.2, CM-6(a), PR.IP-1, PR.PT-3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Remediation script: (show)

Enable cron Service [ref]

rule

The `crond` service is used to execute commands at preconfigured times. It is required by almost all systems to perform necessary maintenance tasks, such as notifying root of system activity. The `crond` service can be enabled with the following command:

```
$ sudo systemctl enable crond.service
```

Rationale:
Due to its usage for maintenance and security-supporting tasks, enabling the cron daemon is essential.

Severity: medium

References: 11, 14, 3, 9, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS06.06, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.9.1.2, CM-6(a), PR.IP-1, PR.PT-3, 5.1.1

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Disable At Service (atd) [ref]

rule

The at and batch commands can be used to schedule tasks that are meant to be executed only once. This allows delayed execution in a manner similar to cron, except that it is not recurring. The daemon atd keeps track of tasks scheduled via at and batch , and executes them at the specified time. The atd service can be disabled with the following command:

```
$ sudo systemctl mask --now atd.service
```

Rationale:

The atd service could be used by an unsophisticated insider to carry out activities outside of a normal login session, which could complicate accountability. Furthermore, the need to schedule tasks with at or batch is not common.

Severity: medium

References: 11, 14, 3, 9, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.05, DSS06.06, CCI-000381, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.9.1.2, CM-7(a), CM-7(b), CM-6(a), PR.IP-1, PR.PT-3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation script: (show)

Remediation script: (show)

Deprecated services [ref]

group

Some deprecated software services impact the overall system security due to their behavior (leak of confidentiality in network exchange, usage as uncontrolled communication channel, risk associated with the service due to its old age, etc.

▼ contains 5 rules

Uninstall the inet-based telnet server [ref]

rule

The inet-based telnet daemon should be uninstalled.

Rationale:

`telnet` allows clear text communications, and does not protect any data transmission between client and server. Any confidential data can be listened and no integrity checking is made.

Severity: high

References: NT007(R03), 11, 12, 14, 15, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.05, DSS06.06, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.11.2.6, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.9.1.2, CM-7(a), CM-7(b), CM-6(a), PR.AC-3, PR.IP-1, PR.PT-3, PR.PT-4, TAMU-CM-1(2), TAMU-CM-7, TAMU-IA-5(3), TAMU-IA-5(3.2), TAMU-SC-8(2), TAMU-SC-8(3), TAMU-SC-13(2), TAMU-SC-13(3)

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Uninstall the nis package [ref]

rule

The support for Yellowpages should not be installed unless it is required.

Rationale:

NIS is the historical SUN service for central account management, more and more replaced by LDAP. NIS does not support efficiently security constraints, ACL, etc. and should not be used.

Severity: low

References: TAMU-CM-1(2), TAMU-CM-7

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Uninstall the ntpdate package [ref]

rule

ntpdate is a historical ntp synchronization client for unixes. It should be uninstalled.

Rationale:
ntpdate is an old not security-compliant ntp client. It should be replaced by modern ntp clients such as ntpd, able to use cryptographic mechanisms integrated in NTP.

Severity: low

References: TAMU-CM-1(2), TAMU-CM-7, TAMU-AU-3(1.1), TAMU-AU-8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Uninstall the ssl compliant telnet server [ref]

rule

The `telnet` daemon, even with ssl support, should be uninstalled.

Rationale:
`telnet` , even with ssl support, should not be installed. When remote shell is required, up-to-date ssh daemon can be used.

Severity: high

References: NT007(R02), 11, 12, 14, 15, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.05, DSS06.06, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.11.2.6, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.9.1.2, CM-7(a), CM-7(b), CM-6(a), PR.AC-3, PR.IP-1, PR.PT-3, PR.PT-4, TAMU-CM-1(2), TAMU-CM-7, TAMU-IA-5(3), TAMU-IA-5(3.2), TAMU-SC-8(2), TAMU-SC-8(3), TAMU-SC-13(2), TAMU-SC-13(3)

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Uninstall the telnet server [ref]

rule

The telnet daemon should be uninstalled.

Rationale:
telnet allows clear text communications, and does not protect any data transmission between client and server. Any confidential data can be listened and no integrity checking is made.'

Severity: high

References: BP28(R1), NT007(R03), 11, 12, 14, 15, 3, 8, 9, APO13.01, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS01.04, DSS05.02, DSS05.03, DSS05.05, DSS06.06, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, A.11.2.6, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.2.1, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.2.1, A.6.2.2, A.9.1.2, CM-7(a), CM-7(b), CM-6(a), PR.AC-3, PR.IP-1, PR.PT-3, PR.PT-4, TAMU-CM-1(2), TAMU-CM-7, TAMU-IA-5(3), TAMU-IA-5(3.2), TAMU-SC-8(2), TAMU-SC-8(3), TAMU-SC-13(2), TAMU-SC-13(3)

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

SSH Server [ref]

group

The SSH protocol is recommended for remote login and remote file transfer. SSH provides confidentiality and integrity for data exchanged between two systems, as well as server authentication, through the use of public key cryptography. The implementation included with the system is called OpenSSH, and more detailed documentation is available from its website, https://www.openssh.com. Its server program is called sshd and provided by the RPM package openssh-server.

▼ contains 19 rules

Configure OpenSSH Server if Necessary [ref]

group

If the system needs to act as an SSH server, then certain changes should be made to the OpenSSH daemon configuration file /etc/ssh/sshd_config. The following recommendations can be applied to this file. See the sshd_config(5) man page for more detailed information.

▼ contains 16 rules

Disable Host-Based Authentication [ref]

rule

SSH's cryptographic host-based authentication is more secure than `.rhosts` authentication. However, it is not recommended that hosts unilaterally trust one another, even within an organization.

To disable host-based authentication, add or correct the following line in `/etc/ssh/sshd_config`:

```
HostbasedAuthentication no
```

Rationale:
SSH trust relationships mean a compromise on one host can allow an attacker to move trivially to other hosts.

Severity: medium

References: 11, 12, 14, 15, 16, 18, 3, 5, 9, 5.5.6, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.04, DSS05.05, DSS05.07, DSS06.03, DSS06.06, 3.1.12, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, 0421, 0422, 0431, 0974, 1173, 1401, 1504, 1505, 1546, 1557, 1558, 1559, 1560, 1561, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.6.1.2, A.7.1.1, A.9.1.2, A.9.2.1, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.2.3, CIP-004-6 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.2, CIP-007-3 R5.2, CIP-007-3 R5.3.1, CIP-007-3 R5.3.2, CIP-007-3 R5.3.3, AC-3, AC-17(a), CM-7(a), CM-7(b), CM-6(a), PR.AC-4, PR.AC-6, PR.IP-1, PR.PT-3, FIA_UAU.1, SRG-OS-000480-GPOS-00229, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204435r603261_rule, 5.3.9

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation script: (show)

Allow Only SSH Protocol 2 [ref]

rule

Only SSH protocol version 2 connections should be permitted. The default setting in `/etc/ssh/sshd_config` is correct, and can be verified by ensuring that the following line appears:

```
Protocol 2
```

> **Warning:** As of `openssh-server` version `7.4` and above, the only protocol supported is version 2, and line
>
> ```
> Protocol 2
> ```
>
> in `/etc/ssh/sshd_config` is not necessary.

Rationale:
SSH protocol version 1 is an insecure implementation of the SSH protocol and has many well-known vulnerability exploits. Exploits of the SSH daemon could provide immediate root access to the system.

Severity: high

References: NT007(R1), 1, 12, 15, 16, 5, 8, 5.5.6, APO13.01, DSS01.04, DSS05.02, DSS05.03, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.1.13, 3.5.4, CCI-000197, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.6, SR 3.1, SR 3.5, SR 3.8, SR 4.1, SR 4.3, SR 5.1, SR 5.2, SR 5.3, SR 7.1, SR 7.6, 0487, 1449, 1506, A.11.2.6, A.13.1.1, A.13.2.1, A.14.1.3, A.18.1.4, A.6.2.1, A.6.2.2, A.7.1.1, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3, CIP-003-8 R4.2, CIP-007-3 R5.1, CIP-007-3 R7.1, CM-6(a), AC-17(a), AC-17(2), IA-5(1)(c), SC-13, MA-4(6), PR.AC-1, PR.AC-3, PR.AC-6, PR.AC-7, PR.PT-4, SRG-OS-000074-GPOS-00042, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, TAMU-IA-5(3), TAMU-IA-5(3.2), TAMU-SC-8(2), TAMU-SC-8(3), TAMU-SC-13(2), TAMU-SC-13(3), SRG-OS-000033-VMM-000140, SV-204594r603261_rule, 5.2.2

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable SSH UseDNS Checks [ref]

rule

To disable dns client checks, add or correct the following line in `/etc/ssh/sshd_config` :

```
UseDNS no
```

Rationale:
Disabling sshd DNS checks slightly weakens your sshd security. However, if DNS resolution fails, SSH logins still work. If you do not disable sshd checks, DNS resolution failures prevent all logins.

Severity: medium

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable SSH Access via Empty Passwords [ref]

rule

To explicitly disallow SSH login from accounts with empty passwords, add or correct the following line in /etc/ssh/sshd_config :

```
PermitEmptyPasswords no
```

Any accounts with empty passwords should be disabled immediately, and PAM configuration should prevent users from being able to assign themselves empty passwords.

Rationale:
Configuring this setting for the SSH daemon provides additional assurance that remote login via SSH will require a password, even in the event of misconfiguration elsewhere.

Severity: high

References: NT007(R17), 11, 12, 13, 14, 15, 16, 18, 3, 5, 9, 5.5.6, APO01.06, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.04, DSS05.05, DSS05.07, DSS06.02, DSS06.03, DSS06.06, 3.1.1, 3.1.5, CCI-000366, CCI-000766, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 5.2, SR 7.6, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.12.1.2, A.12.5.1, A.12.6.2, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.14.2.2, A.14.2.3, A.14.2.4, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.1, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, AC-17(a), CM-7(a), CM-7(b), CM-6(a), PR.AC-4, PR.AC-6, PR.DS-5, PR.IP-1, PR.PT-3, FIA_UAU.1, SRG-OS-000106-GPOS-00053, SRG-OS-000480-GPOS-00229, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-1(4), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204425r603261_rule, 5.3.11

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable GSSAPI Authentication [ref]

rule

Unless needed, SSH should not permit extraneous or unnecessary authentication mechanisms like GSSAPI. To disable GSSAPI authentication, add or correct the following line in the /etc/ssh/sshd_config file:

```
GSSAPIAuthentication no
```

Rationale:
GSSAPI authentication is used to provide additional authentication mechanisms to applications. Allowing GSSAPI authentication through SSH exposes the system's GSSAPI to remote hosts, increasing the attack surface of the system.

Severity: medium

References: 11, 3, 9, BAI10.01, BAI10.02, BAI10.03, BAI10.05, 3.1.12, CCI-000318, CCI-000368, CCI-001812, CCI-001813, CCI-001814, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.4.3.2, 4.3.4.3.3, SR 7.6, 0418, 1055, 1402, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, CM-7(a), CM-7(b), CM-6(a), AC-17(a), PR.IP-1, FTP_ITC_EXT.1, SRG-OS-000364-GPOS-00151, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204598r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable Kerberos Authentication [ref]

rule

Unless needed, SSH should not permit extraneous or unnecessary authentication mechanisms like Kerberos. To disable Kerberos authentication, add or correct the following line in the `/etc/ssh/sshd_config` file:

```
KerberosAuthentication no
```

Rationale:
Kerberos authentication for SSH is often implemented using GSSAPI. If Kerberos is enabled through SSH, the SSH daemon provides a means of access to the system's Kerberos implementation. Vulnerabilities in the system's Kerberos implementations may be subject to exploitation.

Severity: medium

References: 11, 3, 9, BAI10.01, BAI10.02, BAI10.03, BAI10.05, 3.1.12, CCI-000318, CCI-000368, CCI-001812, CCI-001813, CCI-001814, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.4.3.2, 4.3.4.3.3, SR 7.6, 0421, 0422, 0431, 0974, 1173, 1401, 1504, 1505, 1546, 1557, 1558, 1559, 1560, 1561, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, AC-17(a), CM-7(a), CM-7(b), CM-6(a), PR.IP-1, FTP_ITC_EXT.1, SRG-OS-000364-GPOS-00151, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204599r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable SSH Support for .rhosts Files [ref]

rule

SSH can emulate the behavior of the obsolete rsh command in allowing users to enable insecure access to their accounts via `.rhosts` files.

To ensure this behavior is disabled, add or correct the following line in `/etc/ssh/sshd_config`:

```
IgnoreRhosts yes
```

Rationale:
SSH trust relationships mean a compromise on one host can allow an attacker to move trivially to other hosts.

Severity: medium

References: 11, 12, 14, 15, 16, 18, 3, 5, 9, 5.5.6, BAI10.01, BAI10.02, BAI10.03, BAI10.05, DSS05.02, DSS05.04, DSS05.05, DSS05.07, DSS06.03, DSS06.06, 3.1.12, CCI-000366, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.2, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 7.6, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, A.6.1.2, A.7.1.1, A.9.1.2, A.9.2.1, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, AC-17(a), CM-7(a), CM-7(b), CM-6(a), PR.AC-4, PR.AC-6, PR.IP-1, PR.PT-3, FIA_UAU.1, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000107-VMM-000530, SV-204590r603261_rule, 5.3.8

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable SSH Root Login [ref]

rule

The root user should never be allowed to login to a system directly over a network. To disable root login via SSH, add or correct the following line in /etc/ssh/sshd_config :

```
PermitRootLogin no
```

Rationale:
Even though the communications channel may be encrypted, an additional layer of security is gained by extending the policy of not logging directly on as root. In addition, logging in with a user-specific account provides individual accountability of actions performed on the system and also helps to minimize direct attack attempts on root's password.

Severity: medium

References: BP28(R19), NT007(R21), 1, 11, 12, 13, 14, 15, 16, 18, 3, 5, 5.5.6, APO01.06, DSS05.02, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.02, DSS06.03, DSS06.06, DSS06.10, 3.1.1, 3.1.5, CCI-000366, CCI-000770, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.5.3, 4.3.3.5.4, 4.3.3.5.5, 4.3.3.5.6, 4.3.3.5.7, 4.3.3.5.8, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.1, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, SR 1.1, SR 1.10, SR 1.11, SR 1.12, SR 1.13, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.6, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 2.2, SR 2.3, SR 2.4, SR 2.5, SR 2.6, SR 2.7, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.18.1.4, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.1, A.9.4.2, A.9.4.3, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.2.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CIP-007-3 R5.2, CIP-007-3 R5.3.1, CIP-007-3 R5.3.2, CIP-007-3 R5.3.3, AC-6(2), AC-17(a), IA-2, IA-2(5), CM-7(a), CM-7(b), CM-6(a), PR.AC-1, PR.AC-4, PR.AC-6, PR.AC-7, PR.DS-5, PR.PT-3, FIA_UAU.1, SRG-OS-000109-GPOS-00056, SRG-OS-000480-GPOS-00227, TAMU-AC-6(1), TAMU-CM-1(2), TAMU-CM-1(4), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204592r603261_rule, 5.3.10

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Disable X11 Forwarding [ref]

rule

The X11Forwarding parameter provides the ability to tunnel X11 traffic through the connection to enable remote graphic connections. SSH has the capability to encrypt remote X11 connections when SSH's X11Forwarding option is enabled.

To disable X11 Forwarding, add or correct the following line in /etc/ssh/sshd_config :

```
X11Forwarding no
```

Rationale:
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: medium

References: CCI-000366, CM-6(b), SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, SV-204622r603849_rule, 5.3.6

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Do Not Allow SSH Environment Options [ref]

rule

To ensure users are not able to override environment variables of the SSH daemon, add or correct the following line in /etc/ssh/sshd_config :

```
PermitUserEnvironment no
```

Rationale:
SSH environment options potentially allow users to bypass access restriction in some configurations.

Severity: medium

References: 11, 3, 9, 5.5.6, BAI10.01, BAI10.02, BAI10.03, BAI10.05, 3.1.12, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.4.3.2, 4.3.4.3.3, SR 7.6, A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4, AC-17(a), CM-7(a), CM-7(b), CM-6(a), PR.IP-1, SRG-OS-000480-GPOS-00229, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204434r603261_rule, 5.3.12

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Enable Use of Strict Mode Checking [ref]

rule

SSHs StrictModes option checks file and ownership permissions in the user's home directory .ssh folder before accepting login. If world- writable permissions are found, logon is rejected. To enable StrictModes in SSH, add or correct the following line in the /etc/ssh/sshd_config file:

```
StrictModes yes
```

Rationale:
If other users have access to modify user-specific SSH configuration files, they may be able to log into the system as another user.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 3.1.12, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, AC-6, AC-17(a), CM-6(a), PR.AC-4, PR.DS-5, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204600r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Enable SSH Warning Banner [ref] rule

To enable the warning banner and ensure it is consistent across the system, add or correct the following line in /etc/ssh/sshd_config :

```
Banner /etc/issue
```

Another section contains information on how to create an appropriate system-wide warning banner.

Rationale:
The warning message reinforces policy awareness during the logon process and facilitates possible legal action against attackers. Alternatively, systems whose ownership should not be obvious should ensure usage of a banner that does not provide easy attribution.

Severity: medium

References: 1, 12, 15, 16, 5.5.6, DSS05.04, DSS05.10, DSS06.10, 3.1.9, CCI-000048, CCI-000050, CCI-001384, CCI-001385, CCI-001386, CCI-001387, CCI-001388, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, SR 1.1, SR 1.10, SR 1.2, SR 1.5, SR 1.7, SR 1.8, SR 1.9, A.18.1.4, A.9.2.1, A.9.2.4, A.9.3.1, A.9.4.2, A.9.4.3, AC-8(a), AC-8(c), AC-17(a), CM-6(a), PR.AC-7, FTA_TAB.1, SRG-OS-000023-GPOS-00006, SRG-OS-000024-GPOS-00007, SRG-OS-000228-GPOS-00088, TAMU-AC-8(2), TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000023-VMM-000060, SRG-OS-000024-VMM-000070, SV-204580r603261_rule, 5.2.15

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set SSH Idle Timeout Interval [ref]

rule

SSH allows administrators to set an idle timeout interval. After this interval has passed, the idle user will be automatically logged out.

To set an idle timeout interval, edit the following line in /etc/ssh/sshd_config as follows:

```
ClientAliveInterval 900
```

The timeout **interval** is given in seconds. For example, have a timeout of 10 minutes, set **interval** to 600.

If a shorter timeout has already been set for the login shell, that value will preempt any SSH setting made in /etc/ssh/sshd_config. Keep in mind that some processes may stop SSH from correctly detecting that the user is idle.

Warning: SSH disconnecting idle clients will not have desired effect without also configuring ClientAliveCountMax in the SSH service configuration.

Warning: Following conditions may prevent the SSH session to time out:

- Remote processes on the remote machine generates output. As the output has to be transferred over the network to the client, the timeout is reset every time such transfer happens.
- Any scp or sftp activity by the same user to the host resets the timeout.

Rationale:
Terminating an idle ssh session within a short time period reduces the window of opportunity for unauthorized personnel to take control of a management session enabled on the console or console port that has been let unattended.

Severity: medium

References: BP28(R29), 1, 12, 13, 14, 15, 16, 18, 3, 5, 7, 8, 5.5.6, APO13.01, BAI03.01, BAI03.02, BAI03.03, DSS01.03, DSS03.05, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.1.11, CCI-000879, CCI-001133, CCI-002361, 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 6.2, A.12.4.1, A.12.4.3, A.14.1.1, A.14.2.1, A.14.2.5, A.18.1.4, A.6.1.2, A.6.1.5, A.7.1.1, A.9.1.2, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.1, A.9.4.2, A.9.4.3, A.9.4.4, A.9.4.5, CIP-004-6 R2.2.3, CIP-007-3 R5.1, CIP-007-3 R5.2, CIP-007-3 R5.3.1, CIP-007-3 R5.3.2, CIP-007-3 R5.3.3, CM-6(a), AC-17(a), AC-2(5), AC-12, AC-17(a), SC-10, CM-6(a), DE.CM-1, DE.CM-3, PR.AC-1, PR.AC-4, PR.AC-6, PR.AC-7, PR.IP-2, Req-8.1.8, SRG-OS-000126-GPOS-00066, SRG-OS-000163-GPOS-00072, SRG-OS-000279-GPOS-00109, SRG-OS-000395-GPOS-00175, TAMU-CM-1(2), TAMU-CM-7, SRG-OS-000480-VMM-002000, SV-204587r603261_rule, 5.3.16

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set SSH Client Alive Count Max to zero [ref]

rule

The SSH server sends at most `ClientAliveCountMax` messages during a SSH session and waits for a response from the SSH client. The option `ClientAliveInterval` configures timeout after each `ClientAliveCountMax` message. If the SSH server does not receive a response from the client, then the connection is considered idle and terminated. To ensure the SSH idle timeout occurs precisely when the `ClientAliveInterval` is set, set the `ClientAliveCountMax` to value of `0`.

Rationale:
This ensures a user login will be terminated as soon as the `ClientAliveInterval` is reached.

Severity: medium

References: 1, 12, 13, 14, 15, 16, 18, 3, 5, 7, 8, 5.5.6, APO13.01, BAI03.01, BAI03.02, BAI03.03, DSS01.03, DSS03.05, DSS05.04, DSS05.05, DSS05.07, DSS05.10, DSS06.03, DSS06.10, 3.1.11, CCI-000879, CCI-001133, CCI-002361, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.2.2, 4.3.3.5.1, 4.3.3.5.2, 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9, 4.3.3.7.2, 4.3.3.7.3, 4.3.3.7.4, 4.3.4.3.3, SR 1.1, SR 1.10, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 2.1, SR 6.2, A.12.4.1, A.12.4.3, A.14.1.1, A.14.2.1, A.14.2.5, A.18.1.4, A.6.1.2, A.6.1.5, A.7.1.1, A.9.1.2, A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.1, A.9.4.2, A.9.4.3, A.9.4.4, A.9.4.5, CIP-004-6 R2.2.3, CIP-007-3 R5.1, CIP-007-3 R5.2, CIP-007-3 R5.3.1, CIP-007-3 R5.3.2, CIP-007-3 R5.3.3, AC-2(5), AC-12, AC-17(a), SC-10, CM-6(a), DE.CM-1, DE.CM-3, PR.AC-1, PR.AC-4, PR.AC-6, PR.AC-7, PR.IP-2, Req-8.1.8, SRG-OS-000163-GPOS-00072, SRG-OS-000279-GPOS-00109, SRG-OS-000480-VMM-002000, SV-204589r603261_rule, 5.2.12

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Set LogLevel to INFO [ref]

rule

The INFO parameter specifices that record login and logout activity will be logged. To specify the log level in SSH, add or correct the following line in the `/etc/ssh/sshd_config` file:

```
LogLevel INFO
```

Rationale:
SSH provides several logging levels with varying amounts of verbosity. `DEBUG` is specifically not recommended other than strictly for debugging SSH communications since it provides so much data that it is difficult to identify important security information. `INFO` level is the basic level that only records login activity of SSH users. In many situations, such as Incident Response, it is important to determine when a particular user was active on a system. The logout record can eliminate those users who disconnected, which helps narrow the field.

Severity: low

References: AC-17(a), CM-6(a), TAMU-AU-2(1), TAMU-AU-3(1.6), TAMU-CM-1(2), TAMU-CM-7, 5.3.5

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Enable Use of Privilege Separation [ref]

rule

When enabled, SSH will create an unprivileged child process that has the privilege of the authenticated user. To enable privilege separation in SSH, add or correct the following line in the /etc/ssh/sshd_config file:

```
UsePrivilegeSeparation sandbox
```

Rationale:
SSH daemon privilege separation causes the SSH process to drop root privileges when not needed which would decrease the impact of software vulnerabilities in the unprivileged section.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 3.1.12, CCI-000366, 164.308(a)(4)(i), 164.308(b)(1), 164.308(b)(3), 164.310(b), 164.312(e)(1), 164.312(e)(2)(ii), 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, CM-6(a), AC-17(a), AC-6, PR.AC-4, PR.DS-5, SRG-OS-000480-GPOS-00227, TAMU-CM-1(2), TAMU-CM-7, SV-204601r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Install the OpenSSH Server Package [ref]

rule

The openssh-server package should be installed. The openssh-server package can be installed with the following command:

```
$ sudo yum install openssh-server
```

Rationale:
Without protection of the transmitted information, confidentiality, and integrity may be compromised because unprotected communications can be intercepted and either read or altered.

Severity: medium

References: 13, 14, APO01.06, DSS05.02, DSS05.04, DSS05.07, DSS06.02, DSS06.06, CCI-002418, CCI-002420, CCI-002421, CCI-002422, SR 3.1, SR 3.8, SR 4.1, SR 4.2, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CM-6(a), PR.DS-2, PR.DS-5, FIA_UAU.5, FTP_ITC_EXT.1, SRG-OS-000423-GPOS-00187, SRG-OS-000424-GPOS-00188, SRG-OS-000425-GPOS-00189, SRG-OS-000426-GPOS-00190, TAMU-IA-5(3), TAMU-IA-5(3.2), TAMU-SC-8(2), TAMU-SC-8(3), TAMU-SC-13(2), TAMU-SC-13(3), SV-204585r603261_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Remediation Anaconda snippet: (show)

Remediation script: (show)

Verify Permissions on SSH Server Private *_key Key Files [ref]

rule

To properly set the permissions of /etc/ssh/*_key , run the command:

```
$ sudo chmod 0600 /etc/ssh/*_key
```

Rationale:
If an unauthorized user obtains the private SSH host key file, the host could be impersonated.

Severity: medium

References: BP28(R36), 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 3.1.13, 3.13.10, CCI-000366, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, AC-17(a), CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, SRG-OS-000480-GPOS-00227, SV-204597r792834_rule

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Verify Permissions on SSH Server Public *.pub Key Files [ref]

rule

To properly set the permissions of /etc/ssh/*.pub , run the command:

```
$ sudo chmod 0644 /etc/ssh/*.pub
```

Rationale:
If a public host key file is modified by an unauthorized user, the SSH service may be compromised.

Severity: medium

References: 12, 13, 14, 15, 16, 18, 3, 5, APO01.06, DSS05.04, DSS05.07, DSS06.02, 3.1.13, 3.13.10, CCI-000366, 4.3.3.7.3, SR 2.1, SR 5.2, A.10.1.1, A.11.1.4, A.11.1.5, A.11.2.1, A.13.1.1, A.13.1.3, A.13.2.1, A.13.2.3, A.13.2.4, A.14.1.2, A.14.1.3, A.6.1.2, A.7.1.1, A.7.1.2, A.7.3.1, A.8.2.2, A.8.2.3, A.9.1.1, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5, CIP-003-8 R5.1.1, CIP-003-8 R5.3, CIP-004-6 R2.3, CIP-007-3 R2.1, CIP-007-3 R2.2, CIP-007-3 R2.3, CIP-007-3 R5.1, CIP-007-3 R5.1.1, CIP-007-3 R5.1.2, AC-17(a), CM-6(a), AC-6(1), PR.AC-4, PR.DS-5, SRG-OS-000480-GPOS-00227, SV-204596r603261_rule, 5.3.3

Remediation Shell script: (show)

Remediation Ansible snippet: (show)

Remediation Puppet snippet: (show)

Generated using OpenSCAP 1.2.16