CIS AKS Optimized Azure Linux 3 Benchmark

v1.0.0 - 08-01-2025

Terms of Use

Please see the below link for our current terms of use:

https://www.cisecurity.org/cis-securesuite/cis-securesuite-membership-terms-of-use/

For information on referencing and/or citing CIS Benchmarks in 3rd party documentation (including using portions of Benchmark Recommendations) please contact CIS Legal (legalnotices@cisecurity.org) and request guidance on copyright usage.

NOTE: It is ***NEVER*** acceptable to host a CIS Benchmark in ***ANY*** format (PDF, etc.) on a 3rd party (non-CIS owned) site.

Table of Contents

Terms of Use *1*

Table of Contents *2*

Overview *7*

Important Usage Information **7**

Key Stakeholders **7**

Apply the Correct Version of a Benchmark **8**

Exceptions **8**

Remediation **9**

Summary **9**

Target Technology Details **10**

Intended Audience **10**

Consensus Guidance **11**

Typographical Conventions **12**

Recommendation Definitions *13*

Title **13**

Assessment Status **13**

Automated **13**

Manual **13**

Profile **13**

Description **13**

Rationale Statement **13**

Impact Statement **14**

Audit Procedure **14**

Remediation Procedure **14**

Default Value **14**

References **14**

CIS Critical Security Controls® (CIS Controls®) **14**

Additional Information **14**

Profile Definitions **15**

Acknowledgements **16**

Recommendations *17*

1 Initial Setup **17**

1.1 Filesystem **18**

1.1.1 Configure Filesystem Kernel Modules **19**

1.1.1.1 Ensure cramfs kernel module is not available (Automated) 20

1.1.1.2 Ensure freevxfs kernel module is not available (Automated) 23

1.1.1.3 Ensure hfs kernel module is not available (Automated) 26
1.1.1.4 Ensure hfsplus kernel module is not available (Automated) 29
1.1.1.5 Ensure jffs2 kernel module is not available (Automated) 32
1.1.1.6 Ensure unused filesystems kernel modules are not available (Manual) 35
1.1.2 Configure Filesystem Partitions 41
1.1.2.1 Configure /tmp 42
1.1.2.1.1 Ensure /tmp is a separate partition (Automated) 43
1.1.2.1.2 Ensure nodev option set on /tmp partition (Automated) 47
1.1.2.1.3 Ensure nosuid option set on /tmp partition (Automated) 49
1.1.2.2 Configure /dev/shm 51
1.1.2.2.1 Ensure /dev/shm is a separate partition (Automated) 52
1.1.2.2.2 Ensure nodev option set on /dev/shm partition (Automated) 54
1.1.2.2.3 Ensure nosuid option set on /dev/shm partition (Automated) 56
1.2 Package Management 58
1.2.1 Configure Package Repositories 59
1.2.1.1 Ensure GPG keys are configured (Manual) 60
1.2.1.2 Ensure gpgcheck is configured (Automated) 63
1.2.1.3 Ensure TDNF gpgcheck is globally activated (Automated) 65
1.2.1.4 Ensure package manager repositories are configured (Manual) 67
1.3 Configure Additional Process Hardening 69
1.3.1 Ensure address space layout randomization is enabled (Automated) 70
1.3.2 Ensure ptrace_scope is restricted (Automated) 74
1.3.3 Ensure core dump backtraces are disabled (Automated) 79
1.3.4 Ensure core dump storage is disabled (Automated) 82
1.4 Configure Command Line Warning Banners 85
1.4.1 Ensure local login warning banner is configured properly (Automated) 86
1.4.2 Ensure remote login warning banner is configured properly (Automated) 88
1.4.3 Ensure access to /etc/motd is configured (Automated) 90
1.4.4 Ensure access to /etc/issue is configured (Automated) 92
1.4.5 Ensure access to /etc/issue.net is configured (Automated) 94

2 Services 96
2.1 Configure Time Synchronization 97
2.1.1 Ensure time synchronization is in use (Automated) 98
2.1.2 Ensure chrony is configured (Automated) 100
2.2 Configure Special Purpose Services 102
2.2.1 Ensure xinetd is not installed (Automated) 103
2.2.2 Ensure xorg-x11-server-common is not installed (Automated) 105
2.2.3 Ensure avahi is not installed (Automated) 107
2.2.4 Ensure a print server is not installed (Automated) 109
2.2.5 Ensure a dhcp server is not installed (Automated) 111
2.2.6 Ensure a dns server is not installed (Automated) 113
2.2.7 Ensure FTP client is not installed (Automated) 114
2.2.8 Ensure an ftp server is not installed (Automated) 116
2.2.9 Ensure a tftp server is not installed (Automated) 117
2.2.10 Ensure a web server is not installed (Automated) 119
2.2.11 Ensure IMAP and POP3 server is not installed (Automated) 121
2.2.12 Ensure Samba is not installed (Automated) 123
2.2.13 Ensure HTTP Proxy Server is not installed (Automated) 125
2.2.14 Ensure net-snmp is not installed or the snmpd service is not enabled (Automated) 126
2.2.15 Ensure NIS server is not installed (Automated) 128
2.2.16 Ensure telnet-server is not installed (Automated) 130
2.2.17 Ensure mail transfer agent is configured for local-only mode (Automated) 132
2.2.18 Ensure nfs-utils is not installed or the nfs-server service is masked (Automated) 134
2.2.19 Ensure rsync-daemon is not installed or the rsyncd service is masked (Automated) 136
2.3 Service Clients 138

2.3.1 Ensure NIS Client is not installed (Automated) 139
2.3.2 Ensure rsh client is not installed (Automated) 141
2.3.3 Ensure talk client is not installed (Automated) 143
2.3.4 Ensure telnet client is not installed (Automated) 145
2.3.5 Ensure LDAP client is not installed (Automated) 147
2.3.6 Ensure TFTP client is not installed (Automated) 149

3 Network 150
3.1 Configure Network Kernel Parameters 151
3.1.1 Ensure packet redirect sending is disabled (Automated) 152
3.1.2 Ensure bogus icmp responses are ignored (Automated) 157
3.1.3 Ensure broadcast icmp requests are ignored (Automated) 162
3.1.4 Ensure icmp redirects are not accepted (Automated) 167
3.1.5 Ensure secure icmp redirects are not accepted (Automated) 172
3.1.6 Ensure reverse path filtering is enabled (Automated) 177
3.1.7 Ensure source routed packets are not accepted (Automated) 182
3.1.8 Ensure suspicious packets are logged (Automated) 188
3.1.9 Ensure tcp syn cookies is enabled (Automated) 193
3.1.10 Ensure ipv6 router advertisements are not accepted (Automated) 198

4 Host Based Firewall 203
4.1 Configure host based firewall packages 204
4.1.1 Ensure iptables is installed (Automated) 205
4.1.2 Ensure nftables is not in use (Automated) 206
4.1.3 Ensure firewalld is not in use (Automated) 208

5 Access, Authentication and Authorization 211
5.1 Configure time-based job schedulers 212
5.1.1 Ensure cron daemon is enabled (Automated) 213
5.1.2 Ensure permissions on /etc/crontab are configured (Automated) 214
5.1.3 Ensure permissions on /etc/cron.hourly are configured (Automated) 216
5.1.4 Ensure permissions on /etc/cron.daily are configured (Automated) 218
5.1.5 Ensure permissions on /etc/cron.weekly are configured (Automated) 220
5.1.6 Ensure permissions on /etc/cron.monthly are configured (Automated) 222
5.1.7 Ensure permissions on /etc/cron.d are configured (Automated) 224
5.1.8 Ensure cron is restricted to authorized users (Automated) 226
5.1.9 Ensure at is restricted to authorized users (Automated) 228
5.2 Configure SSH Server 230
5.2.1 Ensure access to /etc/ssh/sshd_config is configured (Automated) 232
5.2.2 Ensure access to SSH private host key files is configured (Automated) 235
5.2.3 Ensure access to SSH public host key files is configured (Automated) 239
5.2.4 Ensure sshd Ciphers are configured (Automated) 243
5.2.5 Ensure sshd KexAlgorithms is configured (Automated) 246
5.2.6 Ensure sshd MACs are configured (Automated) 249
5.2.7 Ensure sshd access is configured (Automated) 252
5.2.8 Ensure sshd Banner is configured (Automated) 255
5.2.9 Ensure sshd ClientAliveInterval and ClientAliveCountMax are configured (Automated) 258
5.2.10 Ensure sshd HostbasedAuthentication is disabled (Automated) 261
5.2.11 Ensure sshd IgnoreRhosts is enabled (Automated) 263
5.2.12 Ensure sshd LoginGraceTime is configured (Automated) 265
5.2.13 Ensure sshd LogLevel is configured (Automated) 267
5.2.14 Ensure sshd MaxAuthTries is configured (Automated) 269
5.2.15 Ensure sshd MaxStartups is configured (Automated) 271
5.2.16 Ensure sshd MaxSessions is configured (Automated) 273
5.2.17 Ensure sshd PermitEmptyPasswords is disabled (Automated) 275
5.2.18 Ensure sshd PermitRootLogin is disabled (Automated) 277

5.2.19 Ensure sshd PermitUserEnvironment is disabled (Automated) 279
5.2.20 Ensure sshd UsePAM is enabled (Automated) 281
5.3 Configure privilege escalation 283
5.3.1 Ensure sudo is installed (Automated) 284
5.3.2 Ensure re-authentication for privilege escalation is not disabled globally (Automated) 286
5.3.3 Ensure sudo authentication timeout is configured correctly (Automated) 288
5.4 Configure PAM 290
5.4.1 Ensure password creation requirements are configured (Automated) 291
5.4.2 Ensure lockout for failed password attempts is configured (Automated) 294
5.4.3 Ensure password hashing algorithm is SHA-512 (Automated) 298
5.4.4 Ensure password reuse is limited (Automated) 300
5.5 User Accounts and Environment 302
5.5.1 Set Shadow Password Suite Parameters 303
5.5.1.1 Ensure password expiration is 365 days or less (Automated) 304
5.5.1.2 Ensure minimum days between password changes is configured (Automated) 306
5.5.1.3 Ensure password expiration warning days is 7 or more (Automated) 308
5.5.1.4 Ensure inactive password lock is 30 days or less (Automated) 310
5.5.1.5 Ensure all users last password change date is in the past (Automated) 312
5.5.2 Ensure system accounts are secured (Automated) 314
5.5.3 Ensure default group for the root account is GID 0 (Automated) 318
5.5.4 Ensure default user umask is 027 or more restrictive (Automated) 319

6 Logging and Auditing 324
6.1 System Logging 325
6.1.1 Configure journald 326
6.1.1.1 Configure systemd-journald service 327
6.1.1.1.1 Ensure journald service is active (Automated) 328
6.1.1.1.2 Ensure journald log file access is configured (Manual) 330
6.1.1.1.3 Ensure journald ForwardToSyslog is configured (Automated) 333
6.1.1.1.4 Ensure systemd-journal-remote service is not in use (Automated) 338
6.1.1.1.5 Ensure journald Storage is configured (Automated) 340
6.1.1.1.6 Ensure journald Compress is configured (Automated) 343
6.1.2 Configure rsyslog 347
6.1.2.1 Ensure rsyslog service is enabled and active (Automated) 348
6.1.2.2 Ensure rsyslog log file creation mode is configured (Automated) 350
6.1.2.3 Ensure rsyslog is not configured to receive logs from a remote client (Automated) .. 352
6.1.3 Configure Logfiles 354
6.1.3.1 Ensure access to all logfiles has been configured (Automated) 355
6.2 Ensure logrotate is configured (Manual) 360

7 System Maintenance 362
7.1 Configure system file and directory access 363
7.1.1 Ensure access to /etc/passwd is configured (Automated) 364
7.1.2 Ensure access to /etc/passwd- is configured (Automated) 366
7.1.3 Ensure access to /etc/group is configured (Automated) 368
7.1.4 Ensure access to /etc/group- is configured (Automated) 370
7.1.5 Ensure access to /etc/shadow is configured (Automated) 372
7.1.6 Ensure access to /etc/shadow- is configured (Automated) 374
7.1.7 Ensure access to /etc/gshadow is configured (Automated) 376
7.1.8 Ensure access to /etc/gshadow- is configured (Automated) 378
7.1.9 Ensure access to /etc/shells is configured (Automated) 380
7.1.10 Ensure access to /etc/security/opasswd is configured (Automated) 382
7.1.11 Ensure world writable files and directories are secured (Automated) 384
7.1.12 Ensure no files or directories without an owner and a group exist (Automated) 388
7.2 Local User and Group Settings 391
7.2.1 Ensure accounts in /etc/passwd use shadowed passwords (Automated) 392
7.2.2 Ensure /etc/shadow password fields are not empty (Automated) 395

7.2.3 Ensure all groups in /etc/passwd exist in /etc/group (Automated) 397
7.2.4 Ensure no duplicate UIDs exist (Automated) 399
7.2.5 Ensure no duplicate GIDs exist (Automated) 401
7.2.6 Ensure no duplicate user names exist (Automated) 403
7.2.7 Ensure no duplicate group names exist (Automated) 405
7.2.8 Ensure local interactive user home directories are configured (Automated) 407
7.2.9 Ensure local interactive user dot files access is configured (Automated) 412
Appendix: Summary Table 417
Appendix: CIS Controls v7 IG 1 Mapped Recommendations 427
Appendix: CIS Controls v7 IG 2 Mapped Recommendations 430
Appendix: CIS Controls v7 IG 3 Mapped Recommendations 435
Appendix: CIS Controls v7 Unmapped Recommendations 440
Appendix: CIS Controls v8 IG 1 Mapped Recommendations 441
Appendix: CIS Controls v8 IG 2 Mapped Recommendations 444
Appendix: CIS Controls v8 IG 3 Mapped Recommendations 449
Appendix: CIS Controls v8 Unmapped Recommendations 454
Appendix: Change History 455

Overview

All CIS Benchmarks™ (Benchmarks) focus on technical configuration settings used to maintain and/or increase the security of the addressed technology, and they should be used in **conjunction** with other essential cyber hygiene tasks like:

- Monitoring the base operating system and applications for vulnerabilities and quickly updating with the latest security patches.
- End-point protection (Antivirus software, Endpoint Detection and Response (EDR), etc.).
- Logging and monitoring user and system activity.

In the end, the Benchmarks are designed to be a key **component** of a comprehensive cybersecurity program.

Important Usage Information

All Benchmarks are available free for non-commercial use from the CIS Website. They can be used to manually assess and remediate systems and applications. In lieu of manual assessment and remediation, there are several tools available to assist with assessment:

- CIS Configuration Assessment Tool (CIS-CAT® Pro Assessor)
- CIS Benchmarks™ Certified 3rd Party Tooling

These tools make the hardening process much more scalable for large numbers of systems and applications.

NOTE: Some tooling focuses only on the Benchmark Recommendations that can be fully automated (skipping ones marked **Manual**). It is important that ***ALL*** Recommendations (**Automated** and **Manual**) be addressed since all are important for properly securing systems and are typically in scope for audits.

Key Stakeholders

Cybersecurity is a collaborative effort, and cross functional cooperation is imperative within an organization to discuss, test, and deploy Benchmarks in an effective and efficient way. The Benchmarks are developed to be best practice configuration guidelines applicable to a wide range of use cases. In some organizations, exceptions to specific Recommendations will be needed, and this team should work to prioritize the problematic Recommendations based on several factors like risk, time, cost, and labor. These exceptions should be properly categorized and documented for auditing purposes.

Apply the Correct Version of a Benchmark

Benchmarks are developed and tested for a specific set of products and versions and applying an incorrect Benchmark to a system can cause the resulting pass/fail score to be incorrect. This is due to the assessment of settings that do not apply to the target systems. To assure the correct Benchmark is being assessed:

- **Deploy the Benchmark applicable to the way settings are managed in the environment:** An example of this is the Microsoft Windows family of Benchmarks, which have separate Benchmarks for Group Policy, Intune, and Stand-alone systems based upon how system management is deployed. Applying the wrong Benchmark in this case will give invalid results.

- **Use the most recent version of a Benchmark**: This is true for all Benchmarks, but especially true for cloud technologies. Cloud technologies change frequently and using an older version of a Benchmark may have invalid methods for auditing and remediation.

Exceptions

The guidance items in the Benchmarks are called recommendations and not requirements, and exceptions to some of them are expected and acceptable. The Benchmarks strive to be a secure baseline, or starting point, for a specific technology, with known issues identified during Benchmark development are documented in the Impact section of each Recommendation. In addition, organizational, system specific requirements, or local site policy may require changes as well, or an exception to a Recommendation or group of Recommendations (e.g. A Benchmark could Recommend that a Web server not be installed on the system, but if a system's primary purpose is to function as a Webserver, there should be a documented exception to this Recommendation for that specific server).

In the end, exceptions to some Benchmark Recommendations are common and acceptable, and should be handled as follows:

- The reasons for the exception should be reviewed cross-functionally and be well documented for audit purposes.
- A plan should be developed for mitigating, or eliminating, the exception in the future, if applicable.
- If the organization decides to accept the risk of this exception (not work toward mitigation or elimination), this should be documented for audit purposes.

It is the responsibility of the organization to determine their overall security policy, and which settings are applicable to their unique needs based on the overall risk profile for the organization.

Remediation

CIS has developed Build Kits for many technologies to assist in the automation of hardening systems. Build Kits are designed to correspond to Benchmark's "Remediation" section, which provides the manual remediation steps necessary to make that Recommendation compliant to the Benchmark.

When remediating systems (changing configuration settings on deployed systems as per the Benchmark's Recommendations), please approach this with caution and test thoroughly.

The following is a reasonable remediation approach to follow:

- CIS Build Kits, or internally developed remediation methods should never be applied to production systems without proper testing.
- Proper testing consists of the following:
  - Understand the configuration (including installed applications) of the targeted systems. Various parts of the organization may need different configurations (e.g., software developers vs standard office workers).
  - Read the Impact section of the given Recommendation to help determine if there might be an issue with the targeted systems.
  - Test the configuration changes with representative lab system(s). If issues arise during testing, they can be resolved prior to deploying to any production systems.
  - When testing is complete, initially deploy to a small sub-set of production systems and monitor closely for issues. If there are issues, they can be resolved prior to deploying more broadly.
  - When the initial deployment above is completes successfully, iteratively deploy to additional systems and monitor closely for issues. Repeat this process until the full deployment is complete.

Summary

Using the Benchmarks Certified tools, working as a team with key stakeholders, being selective with exceptions, and being careful with remediation deployment, it is possible to harden large numbers of deployed systems in a cost effective, efficient, and safe manner.

NOTE: As previously stated, the PDF versions of the CIS Benchmarks™ are available for free, non-commercial use on the CIS Website. All other formats of the CIS Benchmarks™ (MS Word, Excel, and Build Kits) are available for CIS SecureSuite® members.

CIS-CAT® Pro is also available to CIS SecureSuite® members.

Target Technology Details

This document provides prescriptive guidance for establishing a secure configuration posture for Azure Kubernetes Service (AKS) nodes running on Azure Linux.
This guide was developed and tested against Azure Kubernetes Service (AKS) nodes, running Azure Linux, on the Azure Compute Platform (ACP), using x86-64, AMD64, and Arm64 platforms.
The guidance within broadly assumes that operations are being performed as the root user, and executed under the default bash version for the applicable distribution. Operations performed using sudo instead of the root user, or executed under another shell, may produce unexpected results, or fail to make the intended changes to the system. Non-root users may not be able to access certain areas of the system, especially after remediation has been performed. It is advisable to verify root users path integrity and the integrity of any programs being run prior to execution of commands and scripts included in this benchmark.
To obtain the latest version of this guide, please visit http://workbench.cisecurity.org. If you have questions, comments, or have identified ways to improve this guide, please write us at feedback@cisecurity.org.

Intended Audience

This benchmark is intended for system and application administrators, security specialists, auditors, help desk, and platform deployment personnel who plan to develop, deploy, assess, or secure solutions that incorporate Azure Kubernetes Service (AKS) nodes running Azure Linux on the Azure Compute Platform (ACP).

Consensus Guidance

This CIS Benchmark™ was created using a consensus review process comprised of a global community of subject matter experts. The process combines real world experience with data-based information to create technology specific guidance to assist users to secure their environments. Consensus participants provide perspective from a diverse set of backgrounds including consulting, software development, audit and compliance, security research, operations, government, and legal.

Each CIS Benchmark undergoes two phases of consensus review. The first phase occurs during initial Benchmark development. During this phase, subject matter experts convene to discuss, create, and test working drafts of the Benchmark. This discussion occurs until consensus has been reached on Benchmark recommendations. The second phase begins after the Benchmark has been published. During this phase, all feedback provided by the Internet community is reviewed by the consensus team for incorporation in the Benchmark. If you are interested in participating in the consensus process, please visit https://workbench.cisecurity.org/.

Typographical Conventions

The following typographical conventions are used throughout this guide:

Convention	Meaning
`Stylized Monospace font`	Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented.
`Monospace font`	Used for inline code, commands, UI/Menu selections or examples. Text should be interpreted exactly as presented.
`<Monospace font in brackets>`	Text set in angle brackets denote a variable requiring substitution for a real value.
Italic font	Used to reference other relevant settings, CIS Benchmarks and/or Benchmark Communities. Also, used to denote the title of a book, article, or other publication.
Bold font	Additional information or caveats things like **Notes**, **Warnings**, or **Cautions** (usually just the word itself and the rest of the text normal).

Recommendation Definitions

The following defines the various components included in a CIS recommendation as applicable. If any of the components are not applicable it will be noted, or the component will not be included in the recommendation.

Title

Concise description for the recommendation's intended configuration.

Assessment Status

An assessment status is included for every recommendation. The assessment status indicates whether the given recommendation can be automated or requires manual steps to implement. Both statuses are equally important and are determined and supported as defined below:

Automated

Represents recommendations for which assessment of a technical control can be fully automated and validated to a pass/fail state. Recommendations will include the necessary information to implement automation.

Manual

Represents recommendations for which assessment of a technical control cannot be fully automated and requires all or some manual steps to validate that the configured state is set as expected. The expected state can vary depending on the environment.

Profile

A collection of recommendations for securing a technology or a supporting platform. Most benchmarks include at least a Level 1 and Level 2 Profile. Level 2 extends Level 1 recommendations and is not a standalone profile. The Profile Definitions section in the benchmark provides the definitions as they pertain to the recommendations included for the technology.

Description

Detailed information pertaining to the setting with which the recommendation is concerned. In some cases, the description will include the recommended value.

Rationale Statement

Detailed reasoning for the recommendation to provide the user a clear and concise understanding on the importance of the recommendation.

Impact Statement

Any security, functionality, or operational consequences that can result from following the recommendation.

Audit Procedure

Systematic instructions for determining if the target system complies with the recommendation.

Remediation Procedure

Systematic instructions for applying recommendations to the target system to bring it into compliance according to the recommendation.

Default Value

Default value for the given setting in this recommendation, if known. If not known, either not configured or not defined will be applied.

References

Additional documentation relative to the recommendation.

CIS Critical Security Controls® (CIS Controls®)

The mapping between a recommendation and the CIS Controls is organized by CIS Controls version, Safeguard, and Implementation Group (IG). The Benchmark in its entirety addresses the CIS Controls safeguards of (v7) “5.1 - Establish Secure Configurations” and (v8) '4.1 - Establish and Maintain a Secure Configuration Process” so individual recommendations will not be mapped to these safeguards.

Additional Information

Supplementary information that does not correspond to any other field but may be useful to the user.

Profile Definitions

The following configuration profiles are defined by this Benchmark:

- **Level 1 - Server**

  Items in this profile intend to:

  - be practical and prudent;
  - provide a clear security benefit; and
  - not inhibit the utility of the technology beyond acceptable means.

  This profile is intended for servers.

Acknowledgements

This Benchmark exemplifies the great things a community of users, vendors, and subject matter experts can accomplish through consensus collaboration. The CIS community thanks the entire consensus team with special recognition to the following individuals who contributed greatly to the creation of this guide:

This benchmark is based upon previous Linux benchmarks published and would not be possible without the contributions provided over the history of all of these benchmarks. The CIS community thanks everyone who has contributed to the Linux benchmarks.

Contributor
Graham Eames
Simon John

Editor
Jonathan Lewis Christopherson
Eric Pinnell
Lynsey Rydberg
Tobias Brick
Randie Bejar

Recommendations

1 Initial Setup

Items in this section are advised for all systems, but may be difficult or require extensive preparation after the initial setup of the system.

1.1 Filesystem

The file system is generally a built-in layer used to handle the data management of the storage.

1.1.1 Configure Filesystem Kernel Modules

Several uncommon filesystem types are supported under Linux. Removing support for unneeded filesystem types reduces the local attack surface of the system. If a filesystem type is not needed it should be disabled. Native Linux file systems are designed to ensure that built-in security controls function as expected. Non-native filesystems can lead to unexpected consequences to both the security and functionality of the system and should be used with caution. Many filesystems are created for niche use cases and are not maintained and supported as the operating systems are updated and patched. Users of non-native filesystems should ensure that there is attention and ongoing support for them, especially in light of frequent operating system changes.

Standard network connectivity and Internet access to cloud storage may make the use of non-standard filesystem formats to directly attach heterogeneous devices much less attractive.

Note: This should not be considered a comprehensive list of filesystems. You may wish to consider additions to those listed here for your environment. For the current available file system modules on the system see `ls /usr/lib/modules/**/kernel/fs | sort -u`.

Start up scripts

Kernel modules loaded directly via `insmod` will ignore what is configured in the relevant `/etc/modprobe.d/*.conf` files. If modules are still being loaded after a reboot whilst having the correctly configured `blacklist` and `install` command, check for `insmod` entries in start up scripts such as `.bashrc`.

Return values

Using `/bin/false` as the command in disabling a particular module serves two purposes; to convey the meaning of the entry to the user and cause a non-zero return value. The latter can be tested for in scripts. Please note that `insmod` will ignore what is configured in the relevant configuration files. The preferred way to load modules is with `modprobe`.

1.1.1.1 Ensure cramfs kernel module is not available (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `cramfs` filesystem type is a compressed read-only Linux filesystem embedded in small footprint systems. A `cramfs` image can be used without having to first decompress the image.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it.

Audit:

Verify the cramfs kernel module is not available on the system **- OR -** has been disabled.
Run the following script to determine if the cramfs kernel module is available on the system:

```
#!/usr/bin/env bash

{
   l_mod_name="cramfs" l_mod_type="fs"
   while IFS= read -r l_mod_path; do
      if [ -d "$l_mod_path/${l_mod_name/-/\/}" ] &&  [ -n "$(ls -A
"$l_mod_path/${l_mod_name/-/\/}")" ]; then
         printf '%s\n' "$l_mod_name exists in $l_mod_path"
      fi
   done < <(readlink -f /usr/lib/modules/**/kernel/$l_mod_type || readlink -f
/lib/modules/**/kernel/$l_mod_type)
}
```

If nothing is returned, the cramfs kernel module is not available on the system and no further audit steps are required.
Note: Some systems may include the cramfs filesystem as part of the kernel opposed to being available as a kernel module. In this case, the above audit will not return anything. This is also considered a passing state.
If anything is returned by the above script, verify the cramfs kernel module is not loaded and not loadable by performing the following:
Run the following command to verify the cramfs kernel module is not loaded:

```
lsmod | grep 'cramfs'
```

Nothing should be returned.
Run the following command to verify the cramfs kernel module is not loadable:

```
modprobe --showconfig | grep -P -- '\b(install|blacklist)\h+cramfs\b'
```

Verify the output includes:

```
blacklist cramfs
 -AND EITHER-
install cramfs /bin/false
 -OR-
install cramfs /bin/true
```

Example output:

```
blacklist cramfs
install cramfs /bin/false
```

Remediation:

Run the following to unload and disable the `cramfs` kernel module.
Run the following commands to unload the `cramfs` kernel module:

```
# modprobe -r cramfs 2>/dev/null
# rmmod cramfs 2>/dev/null
```

Perform the following to disable the `cramfs` kernel module:
Create a file ending in `.conf` with `install cramfs /bin/false` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "install cramfs /bin/false" >> /etc/modprobe.d/cramfs.conf
```

Create a file ending in `.conf` with `blacklist cramfs` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "blacklist cramfs" >> /etc/modprobe.d/cramfs.conf
```

References:

1. CCI-000381
2. NIST SP 800-53: CM-7 a
3. NIST SP 800-53A :: CM-7.1 (ii)
4. STIG ID: RHEL-08-040025 | RULE ID: SV-230498r1069314 | CAT III
5. STIG ID: RHEL-09-231195 | RULE ID: SV-257880r1044951 | CAT III
6. STIG ID: ALMA-09-029940 | RULE ID: SV-269344r1050226 | CAT II

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.1.2 Ensure freevxfs kernel module is not available (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `freevxfs` filesystem type is a free version of the Veritas type filesystem. This is the primary filesystem type for HP-UX operating systems.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it.

Audit:

Verify the freevxfs kernel module is not available on the system or has been disabled. Run the following script to determine if the freevxfs kernel module is available on the system:

```
#!/usr/bin/env bash

{
   l_mod_name="freevxfs" l_mod_type="fs"
   while IFS= read -r l_mod_path; do
      if [ -d "$l_mod_path/${l_mod_name/-/\/}" ] &&  [ -n "$(ls -A
"$l_mod_path/${l_mod_name/-/\/}")" ]; then
         printf '%s\n' "$l_mod_name exists in $l_mod_path"
      fi
   done < <(readlink -f /usr/lib/modules/**/kernel/$l_mod_type || readlink -f
/lib/modules/**/kernel/$l_mod_type)
}
```

If nothing is returned, the freevxfs kernel module is not available on the system and no further audit steps are required.
Note: Some systems may include the freevxfs filesystem as part of the kernel opposed to being available as a kernel module. In this case, the above audit will not return anything. This is also considered a passing state.
If anything is returned, verify the freevxfs kernel module is not loaded and not loadable by performing the following:
Run the following command to verify the freevxfs kernel module is not loaded:

```
# lsmod | grep 'freevxfs'
```

Nothing should be returned.
Run the following command to verify the freevxfs kernel module is not loadable:

```
modprobe --showconfig | grep -P -- '\b(install|blacklist)\h+freevxfs\b'
```

Verify the output includes:

```
blacklist freevxfs
 -AND-
install freevxfs /bin/false
 -OR-
install freevxfs /bin/true
```

Example output:

```
blacklist freevxfs
install freevxfs /bin/false
```

Remediation:

Run the following to unload and disable the `freevxfs` kernel module.
Run the following commands to unload the `freevxfs` kernel module:

```
modprobe -r freevxfs 2>/dev/null
rmmod freevxfs 2>/dev/null
```

Perform the following to disable the `freevxfs` kernel module:
Create a file ending in `.conf` with `install freevxfs /bin/false` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "install freevxfs /bin/false" >>
/etc/modprobe.d/freevxfs.conf
```

Create a file ending in `.conf` with `blacklist freevxfs` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "blacklist freevxfs" >> /etc/modprobe.d/freevxfs.conf
```

References:

1. NIST SP 800-53: CM-7 a
2. NIST SP 800-53A: CM-7.1 (ii)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.1.3 Ensure hfs kernel module is not available (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `hfs` filesystem type is a hierarchical filesystem that allows you to mount Mac OS filesystems.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it.

Audit:

Verify the hfs kernel module is not available on the system or has been disabled.
Run the following script to determine if the hfs kernel module is available on the system:

```
#!/usr/bin/env bash

{
   l_mod_name="hfs" l_mod_type="fs"
   while IFS= read -r l_mod_path; do
      if [ -d "$l_mod_path/${l_mod_name/-/\/}" ] &&  [ -n "$(ls -A
"$l_mod_path/${l_mod_name/-/\/}")" ]; then
         printf '%s\n' "$l_mod_name exists in $l_mod_path"
      fi
   done < <(readlink -f /usr/lib/modules/**/kernel/$l_mod_type || readlink -f
/lib/modules/**/kernel/$l_mod_type)
}
```

If nothing is returned, the hfs kernel module is not available on the system and no further audit steps are required.
Note: Some systems may include the hfs filesystem as part of the kernel opposed to being available as a kernel module. In this case, the above audit will not return anything. This is also considered a passing state.
If anything is returned, verify the hfs kernel module is not loaded and not loadable by performing the following:
Run the following command to verify the hfs kernel module is not loaded:

```
lsmod | grep 'hfs'
```

Nothing should be returned.
Run the following command to verify the hfs kernel module is not loadable:

```
modprobe --showconfig | grep -P -- '\b(install|blacklist)\h+hfs\b'
```

Verify the output includes:

```
blacklist hfs
 -AND-
install hfs /bin/false
 -OR-
install hfs /bin/true
```

Example output:

```
blacklist hfs
install hfs /bin/false
```

Remediation:

Run the following to unload and disable the `hfs` kernel module.
Run the following commands to unload the `hfs` kernel module:

```
modprobe -r hfs 2>/dev/null
rmmod hfs 2>/dev/null
```

Perform the following to disable the `hfs` kernel module:
Create a file ending in `.conf` with `install hfs /bin/false` in the `/etc/modprobe.d/` directory.
Example:

```
# printf '%s\n' "" "install hfs /bin/false" >> /etc/modprobe.d/hfs.conf
```

Create a file ending in `.conf` with `blacklist hfs` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "blacklist hfs" >> /etc/modprobe.d/hfs.conf
```

References:

1. NIST SP 800-53: CM-7 a
2. NIST SP 800-53A :: CM-7.1 (ii)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.1.4 Ensure hfsplus kernel module is not available (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `hfsplus` filesystem type is a hierarchical filesystem designed to replace `hfs` that allows you to mount Mac OS filesystems.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it.

Audit:

Verify the hfsplus kernel module is not available on the system or has been disabled. Run the following script to determine if the hfsplus kernel module is available on the system:

```
#!/usr/bin/env bash

{
   l_mod_name="hfsplus" l_mod_type="fs"
   while IFS= read -r l_mod_path; do
      if [ -d "$l_mod_path/${l_mod_name/-/\/}" ] &&  [ -n "$(ls -A
"$l_mod_path/${l_mod_name/-/\/}")" ]; then
         printf '%s\n' "$l_mod_name exists in $l_mod_path"
      fi
   done < <(readlink -f /usr/lib/modules/**/kernel/$l_mod_type || readlink -f
/lib/modules/**/kernel/$l_mod_type)
}
```

If nothing is returned, the hfsplus kernel module is not available on the system and no further audit steps are required.
Note: Some systems may include the hfsplus filesystem as part of the kernel opposed to being available as a kernel module. In this case, the above audit will not return anything. This is also considered a passing state.
If anything is returned, verify the hfsplus kernel module is not loaded and not loadable by performing the following:
Run the following command to verify the hfsplus kernel module is not loaded:

```
lsmod | grep 'hfsplus'
```

Nothing should be returned.
Run the following command to verify the hfsplus kernel module is not loadable:

```
modprobe --showconfig | grep -P -- '\b(install|blacklist)\h+hfsplus\b'
```

Verify the output includes:

```
blacklist hfsplus
 -AND-
install hfsplus /bin/false
 -OR-
install hfsplus /bin/true
```

Example output:

```
blacklist hfsplus
install hfsplus /bin/false
```

Remediation:

Run the following to unload and disable the `hfsplus` kernel module.
Run the following commands to unload the `hfsplus` kernel module:

```
modprobe -r hfsplus 2>/dev/null
rmmod hfsplus 2>/dev/null
```

Perform the following to disable the `hfsplus` kernel module:
Create a file ending in `.conf` with `install hfsplus /bin/false` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "install hfsplus /bin/false" >> /etc/modprobe.d/hfsplus.conf
```

Create a file ending in `.conf` with `blacklist hfsplus` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "blacklist hfsplus" >> /etc/modprobe.d/hfsplus.conf
```

References:

1. NIST SP 800-53 Rev. 5: CM-7 a
2. NIST SP 800-53A :: CM-7.1 (ii)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.1.5 Ensure jffs2 kernel module is not available (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `jffs2` (journaling flash filesystem 2) filesystem type is a log-structured filesystem used in flash memory devices.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it.

Audit:

Verify the jffs2 kernel module is not available on the system or has been disabled. Run the following script to determine if the jffs2 kernel module is available on the system:

```
#!/usr/bin/env bash

{
   l_mod_name="jffs2" l_mod_type="fs"
   while IFS= read -r l_mod_path; do
      if [ -d "$l_mod_path/${l_mod_name/-/\/}" ] &&  [ -n "$(ls -A
"$l_mod_path/${l_mod_name/-/\/}")" ]; then
         printf '%s\n' "$l_mod_name exists in $l_mod_path"
      fi
   done < <(readlink -f /usr/lib/modules/**/kernel/$l_mod_type || readlink -f
/lib/modules/**/kernel/$l_mod_type)
}
```

If nothing is returned, the jffs2 kernel module is not available on the system and no further audit steps are required.
Note: Some systems may include the jffs2 filesystem as part of the kernel opposed to being available as a kernel module. In this case, the above audit will not return anything. This is also considered a passing state.
If anything is returned, verify the jffs2 kernel module is not loaded and not loadable by performing the following:
Run the following command to verify the jffs2 kernel module is not loaded:

```
lsmod | grep 'jffs2'
```

Nothing should be returned.
Run the following command to verify the jffs2 kernel module is not loadable:

```
modprobe --showconfig | grep -P -- '\b(install|blacklist)\h+jffs2\b'
```

Verify the output includes:

```
blacklist jffs2
 -AND-
install jffs2 /bin/false
 -OR-
install jffs2 /bin/true
```

Example output:

```
blacklist jffs2
install jffs2 /bin/false
```

Remediation:

Run the following to unload and disable the `jffs2` kernel module.
Run the following commands to unload the `jffs2` kernel module:

```
modprobe -r jffs2 2>/dev/null
rmmod jffs2 2>/dev/null
```

Perform the following to disable the `jffs2` kernel module:
Create a file ending in `.conf` with `install jffs2 /bin/false` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "install jffs2 /bin/false" >> /etc/modprobe.d/jffs2.conf
```

Create a file ending in `.conf` with `blacklist jffs2` in the `/etc/modprobe.d/` directory.
Example:

```
printf '%s\n' "" "blacklist jffs2" >> /etc/modprobe.d/jffs2.conf
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.1.6 Ensure unused filesystems kernel modules are not available (Manual)

Profile Applicability:

- Level 1 - Server

Description:

Filesystem kernel modules are pieces of code that can be dynamically loaded into the Linux kernel to extend its filesystem capabilities, or so-called base kernel, of an operating system. Filesystem kernel modules are typically used to add support for new hardware (as device drivers), or for adding system calls.

Rationale:

While loadable filesystem kernel modules are a convenient method of modifying the running kernel, this can be abused by attackers on a compromised system to prevent detection of their processes or files, allowing them to maintain control over the system. Many rootkits make use of loadable filesystem kernel modules in this way.

Removing support for unneeded filesystem types reduces the local attack surface of the system. If this filesystem type is not needed, disable it. The following filesystem kernel modules have known CVE's and should be made unavailable if no dependencies exist:

- `afs` - CVE-2022-37402
- `ceph` - CVE-2022-0670
- `cifs` - CVE-2022-29869
- `exfat` CVE-2022-29973
- `ext` CVE-2022-1184
- `fat` CVE-2022-22043
- `fscache` CVE-2022-3630
- `fuse` CVE-2023-0386
- `gfs2` CVE-2023-3212
- `nfs_common` CVE-2023-6660
- `nfsd` CVE-2022-43945
- `smbfs_common` CVE-2022-2585

Impact:

This list may be quite extensive and covering all edges cases is difficult. Therefore, it's crucial to carefully consider the implications and dependencies before making any changes to the filesystem kernel module configurations.

Audit:

Run the following script to:

- Look at the filesystem kernel modules available to the currently running kernel.
- Exclude mounted filesystem kernel modules that don't currently have a CVE.
- List filesystem kernel modules that are not fully disabled, or are loaded into the kernel.

Review the generated output.

```
#! /usr/bin/env bash

{
   a_output=(); a_output2=(); a_modprope_config=(); a_excluded=(); a_available_modules=()
   a_ignore=("xfs" "vfat" "ext2" "ext3" "ext4")
   a_cve_exists=("afs" "ceph" "cifs" "exfat" "ext" "fat" "fscache" "fuse" "gfs2" "nfs_common"
"nfsd" "smbfs_common")
   f_module_chk()
   {
      l_out2=""; grep -Pq -- "\b$l_mod_name\b" <<< "${a_cve_exists[*]}" && l_out2=" <- CVE
exists!"
      if ! grep -Pq -- '\bblacklist\h+'"$l_mod_name"'\b' <<< "${a_modprope_config[*]}"; then
         a_output2+=("  - Kernel module: \"$l_mod_name\" is not fully disabled $l_out2")
      elif ! grep -Pq -- '\binstall\h+'"$l_mod_name"'\h+(\/usr)?\/bin\/(false|true)\b' <<<
"${a_modprope_config[*]}"; then
         a_output2+=("  - Kernel module: \"$l_mod_name\" is not fully disabled $l_out2")
      fi
      if lsmod | grep "$l_mod_name" &> /dev/null; then # Check if the module is currently loaded
         l_output2+=("  - Kernel module: \"$l_mod_name\" is loaded" "")
      fi
   }
   while IFS= read -r -d $'\0' l_module_dir; do
      a_available_modules+=("$(basename "$l_module_dir")")
   done < <(find "$(readlink -f /usr/lib/modules/"$(uname -r)"/kernel/fs || readlink -f
/lib/modules/"$(uname -r)"/kernel/fs)" -mindepth 1 -maxdepth 1 -type d ! -empty -print0)
   while IFS= read -r l_exclude; do
      if grep -Pq -- "\b$l_exclude\b" <<< "${a_cve_exists[*]}"; then
         a_output2+=("  - ** WARNING: kernel module: \"$l_exclude\" has a CVE and is currently
mounted! **")
      elif
         grep -Pq -- "\b$l_exclude\b" <<< "${a_available_modules[*]}"; then
         a_output+=("  - Kernel module: \"$l_exclude\" is currently mounted - do NOT unload or
disable")
      fi
      ! grep -Pq -- "\b$l_exclude\b" <<< "${a_ignore[*]}" && a_ignore+=("$l_exclude")
   done < <(findmnt -knD | awk '{print $2}' | sort -u)
   while IFS= read -r l_config; do
      a_modprope_config+=("$l_config")
   done < <(modprobe --showconfig | grep -P '^\h*(blacklist|install)')
   for l_mod_name in "${a_available_modules[@]}"; do # Iterate over all filesystem modules
      [[ "$l_mod_name" =~ overlay ]] && l_mod_name="${l_mod_name::-2}"
      if grep -Pq -- "\b$l_mod_name\b" <<< "${a_ignore[*]}"; then
         a_excluded+=(" - Kernel module: \"$l_mod_name\"")
      else
         f_module_chk
      fi
   done
   [ "${#a_excluded[@]}" -gt 0 ] && printf '%s\n' "" " -- INFO --" \
   "The following intentionally skipped" \
    "${a_excluded[@]}"
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "  - No unused filesystem kernel modules are enabled" "${a_output[@]}" ""
   else
      printf '%s\n' "" "-- Audit Result: --" "  ** REVIEW the following **" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "-- Correctly set: --" "${a_output[@]}" ""
   fi
}
```

WARNING: disabling or denylisting filesystem modules that are in use on the system may be FATAL. It is extremely important to thoroughly review this list.

Remediation:

- IF - the module is available in the running kernel:

- Unload the filesystem kernel module from the kernel
- Create a file ending in `.conf` with install filesystem kernel modules `/bin/false` in the `/etc/modprobe.d/` directory
- Create a file ending in `.conf` with deny list filesystem kernel modules in the `/etc/modprobe.d/` directory

WARNING: unloading, disabling or denylisting filesystem modules that are in use on the system maybe FATAL. It is extremely important to thoroughly review the filesystems returned by the audit before following the remediation procedure.
Example of unloading the `gfs2` kernel module:

```
modprobe -r gfs2 2>/dev/null
rmmod gfs2 2>/dev/null
```

Example of fully disabling the `gfs2` kernel module:

```
printf '%s\n' "" "blacklist gfs2" "install gfs2 /bin/false" >>
/etc/modprobe.d/gfs2.conf
```

Note:

- Disabling a kernel module by modifying the command above for each unused filesystem kernel module
- The example `gfs2` must be updated with the appropriate module name for the command or example script bellow to run correctly.

Below is an example script that can be modified to use on various filesystem kernel modules manual remediation process:
Example Script

```
#!/usr/bin/env bash

{
   a_output2=(); a_output3=(); l_dl="" # Initialize arrays and clear
variables
   l_mod_name="gfs2" # set module name
   l_mod_type="fs" # set module type
   l_mod_path="$(readlink -f /usr/lib/modules/**/kernel/$l_mod_type ||
readlink -f /lib/modules/**/kernel/$l_mod_type)"
   f_module_fix()
   {
      l_dl="y" # Set to ignore duplicate checks
      a_showconfig=() # Create array with modprobe output
      while IFS= read -r l_showconfig; do
         a_showconfig+=("$l_showconfig")
      done < <(modprobe --showconfig | grep -P --
'\b(install|blacklist)\h+'"${l_mod_name//-/_}"'\b')
      if  lsmod | grep "$l_mod_name" &> /dev/null; then # Check if the module
is currently loaded
         a_output2+=(" - unloading kernel module: \"$l_mod_name\"")
         modprobe -r "$l_mod_name" 2>/dev/null; rmmod "$l_mod_name"
2>/dev/null
      fi
      if ! grep -Pq -- '\binstall\h+'"${l_mod_name//-
/_}"'\h+(\/usr)?\/bin\/(true|false)\b' <<< "${a_showconfig[*]}"; then
         a_output2+=(" - setting kernel module: \"$l_mod_name\" to
\"$(readlink -f /bin/false)\"")
         printf '%s\n' "install $l_mod_name $(readlink -f /bin/false)" >>
/etc/modprobe.d/"$l_mod_name".conf
      fi
      if ! grep -Pq -- '\bblacklist\h+'"${l_mod_name//-/_}"'\b' <<<
"${a_showconfig[*]}"; then
         a_output2+=(" - denylisting kernel module: \"$l_mod_name\"")
         printf '%s\n' "blacklist $l_mod_name" >>
/etc/modprobe.d/"$l_mod_name".conf
      fi
   }
   for l_mod_base_directory in $l_mod_path; do # Check if the module exists
on the system
      if [ -d "$l_mod_base_directory/${l_mod_name/-/\/}" ] && [ -n "$(ls -A
"$l_mod_base_directory/${l_mod_name/-/\/}")" ]; then
         a_output3+=("  - \"$l_mod_base_directory\"")
         [[ "$l_mod_name" =~ overlay ]] && l_mod_name="${l_mod_name::-2}"
         [ "$l_dl" != "y" ] && f_module_fix
      else
         echo -e " - kernel module: \"$l_mod_name\" doesn't exist in
\"$l_mod_base_directory\""
      fi
   done
   [ "${#a_output3[@]}" -gt 0 ] && printf '%s\n' "" " -- INFO --" " - module:
\"$l_mod_name\" exists in:" "${a_output3[@]}"
   [ "${#a_output2[@]}" -gt 0 ] && printf '%s\n' "" "${a_output2[@]}" ||
printf '%s\n' "" " - No changes needed"
   printf '%s\n' "" " - remediation of kernel module: \"$l_mod_name\"
complete" ""
}
```

References:

1. https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=filesystem

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.2 Configure Filesystem Partitions

Directories that are used for system-wide functions can be further protected by placing them on separate partitions. This provides protection for resource exhaustion and enables the use of mounting options that are applicable to the directory's intended use. Users' data can be stored on separate partitions and have stricter mount options. A user partition is a filesystem that has been established for use by the users and does not contain software for system operations.

The recommendations in this section are easier to perform during initial system installation. If the system is already installed, it is recommended that a full backup be performed before repartitioning the system.

Note:

-IF- you are repartitioning a system that has already been installed (This may require the system to be in single-user mode):

- Mount the new partition to a temporary mountpoint e.g. `mount /dev/sda2 /mnt`
- Copy data from the original partition to the new partition. e.g. `cp -a /var/tmp/* /mnt`
- Verify that all data is present on the new partition. e.g. `ls -la /mnt`
- Unmount the new partition. e.g. `umount /mnt`
- Remove the data from the original directory that was in the old partition. e.g. `rm -Rf /var/tmp/*` Otherwise it will still consume space in the old partition that will be masked when the new filesystem is mounted.
- Mount the new partition to the desired mountpoint. e.g. `mount /dev/sda2 /var/tmp`
- Update `/etc/fstab` with the new mountpoint. e.g. `/dev/sda2 /var/tmp xfs defaults,rw,nosuid,nodev,noexec,relatime 0 0`

1.1.2.1 Configure /tmp

The `/tmp` directory is a world-writable directory used to store data used by the system and user applications for a short period of time. This data should have no expectation of surviving a reboot, as this directory is intended to be emptied after each reboot.

1.1.2.1.1 Ensure /tmp is a separate partition (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/tmp` directory is a world-writable directory used for temporary storage by all users and some applications.

- IF - an entry for `/tmp` exists in `/etc/fstab` it will take precedence over entries in systemd default unit file.

Note: In an environment where the main system is diskless and connected to iSCSI, entries in `/etc/fstab` may not take precedence.

`/tmp` can be configured to use `tmpfs`.

`tmpfs` puts everything into the kernel internal caches. It grows and shrinks to accommodate the files it contains and is able to swap unneeded pages out to swap space. It has maximum size limits which can be adjusted on the fly via `mount -o remount`.

Since `tmpfs` lives completely in the page cache and on swap, all `tmpfs` pages will be shown as "Shmem" in `/proc/meminfo` and "Shared" in `free`. Notice that these counters also include shared memory. The most reliable way to get the count is using `df` and `du`.

`tmpfs` has three mount options for sizing:

- `size`: The limit of allocated bytes for this `tmpfs` instance. The default is half of your physical RAM without swap. If you oversize your `tmpfs` instances the machine will deadlock since the OOM handler will not be able to free that memory.
- `nr_blocks`: The same as size, but in blocks of PAGE_SIZE.
- `nr_inodes`: The maximum number of inodes for this instance. The default is half of the number of your physical RAM pages, or (on a machine with highmem) the number of lowmem RAM pages, whichever is the lower.

These parameters accept a suffix k, m or g and can be changed on remount. The size parameter also accepts a suffix % to limit this `tmpfs` instance to that percentage of your physical RAM. The default, when neither `size` nor `nr_blocks` is specified, is `size=50%`.

Rationale:

Making `/tmp` its own file system allows an administrator to set additional mount options such as the `noexec` option on the mount, making `/tmp` useless for an attacker to install executable code. It would also prevent an attacker from establishing a hard link to a system `setuid` program and waiting for it to be updated. Once the program was updated, the hard link would be broken, and the attacker would have his own copy of the program. If the program happened to have a security vulnerability, the attacker could continue to exploit the known flaw.

This can be accomplished by either mounting `tmpfs` to `/tmp`, or creating a separate partition for `/tmp`.

Impact:

By design, files saved to `/tmp` should have no expectation of surviving a reboot of the system. `tmpfs` is ram based and all files stored to `tmpfs` will be lost when the system is rebooted.

If files need to be persistent through a reboot, they should be saved to `/var/tmp` not `/tmp`.

Since the `/tmp` directory is intended to be world-writable, there is a risk of resource exhaustion if it is not bound to `tmpfs` or a separate partition.

Running out of `/tmp` space is a problem regardless of what kind of filesystem lies under it, but in a configuration where `/tmp` is not a separate file system it will essentially have the whole disk available, as the default installation only creates a single `/` partition. On the other hand, a RAM-based `/tmp` (as with `tmpfs`) will almost certainly be much smaller, which can lead to applications filling up the filesystem much more easily. Another alternative is to create a dedicated partition for `/tmp` from a separate volume or disk. One of the downsides of a disk-based dedicated partition is that it will be slower than `tmpfs` which is RAM-based.

Audit:

Run the following command and verify the output shows that `/tmp` is mounted. Particular requirements pertaining to mount options are covered in ensuing sections.

```
# findmnt -kn /tmp
```

Example output:

```
/tmp   tmpfs  tmpfs  rw,nosuid,nodev,noexec
```

Ensure that systemd will mount the `/tmp` partition at boot time.

```
# systemctl is-enabled tmp.mount
```

Example output:

```
generated
```

Verify output is not `masked` or `disabled`.
Note: By default, systemd will output `generated` if there is an entry in `/etc/fstab` for `/tmp`. This just means systemd will use the entry in `/etc/fstab` instead of its default unit file configuration for `/tmp`.

Remediation:

First ensure that systemd is correctly configured to ensure that `/tmp` will be mounted at boot time.

```
# systemctl unmask tmp.mount
```

For specific configuration requirements of the `/tmp` mount for your environment, modify `/etc/fstab`.
Example of using `tmpfs` with specific mount options:

```
tmpfs   /tmp    tmpfs     defaults,rw,nosuid,nodev,noexec,relatime,size=2G  0
0
```

Note: the `size=2G` is an example of setting a specific size for `tmpfs`.
Example of using a volume or disk with specific mount options. The source location of the volume or disk will vary depending on your environment:

```
<device> /tmp    <fstype>    defaults,nodev,nosuid,noexec   0 0
```

References:

1. https://www.freedesktop.org/wiki/Software/systemd/APIFileSystems/
2. https://www.freedesktop.org/software/systemd/man/systemd-fstab-generator.html
3. https://www.kernel.org/doc/Documentation/filesystems/tmpfs.txt
4. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.2.1.2 Ensure nodev option set on /tmp partition (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `nodev` mount option specifies that the filesystem cannot contain special devices.

Rationale:

Since the `/tmp` filesystem is not intended to support devices, set this option to ensure that users cannot create block or character special devices in `/tmp`.

Audit:

- IF - a separate partition exists for `/tmp`, verify that the `nodev` option is set.
Run the following command to verify that the `nodev` mount option is set.
Example:

```
# findmnt -kn /tmp | grep -v nodev

Nothing should be returned
```

Remediation:

- IF - a separate partition exists for `/tmp`.
Edit the `/etc/fstab` file and add `nodev` to the fourth field (mounting options) for the `/tmp` partition.
Example:

```
<device> /tmp    <fstype>     defaults,rw,nosuid,nodev,noexec,relatime  0 0
```

Run the following command to remount `/tmp` with the configured options:

```
# mount -o remount /tmp
```

References:

1. See the fstab(5) manual page for more information.
2. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.2.1.3 Ensure nosuid option set on /tmp partition (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `nosuid` mount option specifies that the filesystem cannot contain `setuid` files.

Rationale:

Since the `/tmp` filesystem is only intended for temporary file storage, set this option to ensure that users cannot create `setuid` files in `/tmp`.

Audit:

- IF - a separate partition exists for `/tmp`, verify that the `nosuid` option is set.
Run the following command to verify that the `nosuid` mount option is set.
Example:

```
# findmnt -kn /tmp | grep -v nosuid

Nothing should be returned
```

Remediation:

- IF - a separate partition exists for `/tmp`.
Edit the `/etc/fstab` file and add `nosuid` to the fourth field (mounting options) for the `/tmp` partition.
Example:

```
<device> /tmp    <fstype>     defaults,rw,nosuid,nodev,noexec,relatime  0 0
```

Run the following command to remount `/tmp` with the configured options:

```
# mount -o remount /tmp
```

References:

1. See the fstab(5) manual page for more information.
2. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

1.1.2.2 Configure /dev/shm

The `/dev/shm` directory is a world-writable directory that can function as shared memory that facilitates inter process communication (IPC)

1.1.2.2.1 Ensure /dev/shm is a separate partition (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/dev/shm` directory is a world-writable directory that can function as shared memory that facilitates inter process communication (IPC).

Rationale:

Making `/dev/shm` its own file system allows an administrator to set additional mount options such as the `noexec` option on the mount, making `/dev/shm` useless for an attacker to install executable code. It would also prevent an attacker from establishing a hard link to a system `setuid` program and waiting for it to be updated. Once the program was updated, the hard link would be broken and the attacker would have his own copy of the program. If the program happened to have a security vulnerability, the attacker could continue to exploit the known flaw.

This can be accomplished by mounting `tmpfs` to `/dev/shm`.

Impact:

Since the `/dev/shm` directory is intended to be world-writable, there is a risk of resource exhaustion if it is not bound to a separate partition.

`/dev/shm` utilizing `tmpfs` can be resized using the `size={size}` parameter in the relevant entry in `/etc/fstab`.

Audit:

- IF - `/dev/shm` is to be used on the system, run the following command and verify the output shows that `/dev/shm` is mounted. Particular requirements pertaining to mount options are covered in ensuing sections.

```
# findmnt -kn /dev/shm
```

Example output:

```
/dev/shm    tmpfs   tmpfs   rw,nosuid,nodev,noexec,relatime,seclabel
```

Remediation:

For specific configuration requirements of the `/dev/shm` mount for your environment, modify `/etc/fstab`.
Example:

```
tmpfs   /dev/shm        tmpfs
defaults,rw,nosuid,nodev,noexec,relatime,size=2G  0 0
```

References:

1. https://www.freedesktop.org/wiki/Software/systemd/APIFileSystems/
2. https://www.freedesktop.org/software/systemd/man/systemd-fstab-generator.html
3. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.1.2.2.2 Ensure nodev option set on /dev/shm partition (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The nodev mount option specifies that the filesystem cannot contain special devices.

Rationale:

Since the /dev/shm filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create special devices in /dev/shm partitions.

Audit:

- IF - a separate partition exists for /dev/shm, verify that the nodev option is set.

```
# findmnt -kn /dev/shm | grep -v 'nodev'

Nothing should be returned
```

Remediation:

- IF - a separate partition exists for /dev/shm.
Edit the /etc/fstab file and add nodev to the fourth field (mounting options) for the /dev/shm partition. See the fstab(5) manual page for more information.
Example:

```
tmpfs /dev/shm    tmpfs     defaults,rw,nosuid,nodev,noexec,relatime  0 0
```

Run the following command to remount /dev/shm with the configured options:

```
# mount -o remount /dev/shm
```

Note: It is recommended to use tmpfs as the device/filesystem type as /dev/shm is used as shared memory space by applications.

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

Additional Information:

Some distributions mount /dev/shm through other means and require /dev/shm to be added to /etc/fstab even though it is already being mounted on boot. Others may configure /dev/shm in other locations and may override /etc/fstab configuration. Consult the documentation appropriate for your distribution.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

1.1.2.2.3 Ensure nosuid option set on /dev/shm partition (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `nosuid` mount option specifies that the filesystem cannot contain `setuid` files.

Rationale:

Setting this option on a file system prevents users from introducing privileged programs onto the system and allowing non-root users to execute them.

Audit:

- IF - a separate partition exists for `/dev/shm`, verify that the `nosuid` option is set.

```
# findmnt -kn /dev/shm | grep -v 'nosuid'

Nothing should be returned
```

Remediation:

- IF - a separate partition exists for `/dev/shm`.
Edit the `/etc/fstab` file and add `nosuid` to the fourth field (mounting options) for the `/dev/shm` partition. See the `fstab(5)` manual page for more information.
Example:

```
tmpfs /dev/shm    tmpfs     defaults,rw,nosuid,nodev,noexec,relatime  0 0
```

Run the following command to remount `/dev/shm` with the configured options:

```
# mount -o remount /dev/shm
```

Note: It is recommended to use `tmpfs` as the device/filesystem type as `/dev/shm` is used as shared memory space by applications.

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

Additional Information:

Some distributions mount `/dev/shm` through other means and require `/dev/shm` to be added to `/etc/fstab` even though it is already being mounted on boot. Others may configure `/dev/shm` in other locations and may override `/etc/fstab` configuration. Consult the documentation appropriate for your distribution.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

1.2 Package Management

Patch management procedures may vary widely between enterprises. Large enterprises may choose to install a local updates server that can be used in place of their distributions servers, whereas a single deployment of a system may prefer to get updates directly. Updates can be performed automatically or manually, depending on the site's policy for patch management. Organizations may prefer to test patches against their environment on a non-production system before rolling out to production.

Outdated software is vulnerable to cyber criminals and hackers. Software updates help reduce the risk to your organization. The release of software update notes often reveal the patched exploitable entry points to the public. Public knowledge of these exploits can leave your organization more vulnerable to malicious actors attempting to gain entry to your system's data.

Software updates often offer new and improved features and speed enhancements.

For the purpose of this benchmark, the requirement is to ensure that a patch management process is defined and maintained, the specifics of which are left to the organization.

1.2.1 Configure Package Repositories

Patch management procedures may vary widely between enterprises. Large enterprises may choose to install a local updates server that can be used in place of their distributions servers, whereas a single deployment of a system may prefer to get updates directly. Updates can be performed automatically or manually, depending on the site's policy for patch management. Organizations may prefer to test patches against their environment on a non-production system before rolling out to production.

Outdated software is vulnerable to cyber criminals and hackers. Software updates help reduce the risk to your organization. The release of software update notes often reveals the patched exploitable entry points to the public. Public knowledge of these exploits can leave your organization more vulnerable to malicious actors attempting to gain access to your system's data.

Note: Creation of an appropriate patch management policy is left to the organization.

1.2.1.1 Ensure GPG keys are configured (Manual)

Profile Applicability:

- Level 1 - Server

Description:

The RPM Package Manager implements GPG key signing to verify package integrity during and after installation.

Rationale:

It is important to ensure that updates are obtained from a valid source to protect against spoofing that could lead to the inadvertent installation of malware on the system. To this end, verify that GPG keys are configured correctly for your system.

Audit:

List all GPG key URLs
Each repository should have a `gpgkey` with a URL pointing to the location of the GPG key, either local or remote.

```
# grep -r gpgkey /etc/yum.repos.d/* /etc/dnf/dnf.conf
```

List installed GPG keys
Run the following command to list the currently installed keys. These are the active keys used for verification and installation of RPMs. The packages are fake, they are generated on the fly by `tdnf` or `rpm` during the import of keys from the URL specified in the repository configuration.
Example:

```
# for RPM_PACKAGE in $(rpm -q gpg-pubkey); do
  echo "RPM: ${RPM_PACKAGE}"
  RPM_SUMMARY=$(rpm -q --queryformat "%{SUMMARY}" "${RPM_PACKAGE}")
  RPM_PACKAGER=$(rpm -q --queryformat "%{PACKAGER}" "${RPM_PACKAGE}")
  RPM_DATE=$(date +%Y-%m-%d -d "1970-1-1+$((0x$(rpm -q --queryformat
"%{RELEASE}" "${RPM_PACKAGE}") ))sec")
  RPM_KEY_ID=$(rpm -q --queryformat "%{VERSION}" "${RPM_PACKAGE}")
  echo "Packager: ${RPM_PACKAGER}
Summary: ${RPM_SUMMARY}
Creation date: ${RPM_DATE}
Key ID: ${RPM_KEY_ID}
"
done

RPM: gpg-pubkey-9db62fb1-59920156
Packager: Fedora 28 (28) <fedora-28@fedoraproject.org>
Summary: gpg(Fedora 28 (28) <fedora-28@fedoraproject.org>)
Creation date: 2017-08-14
Key ID: 9db62fb1

RPM: gpg-pubkey-09eab3f2-595fbba3
Packager: RPM Fusion free repository for Fedora (28) <rpmfusion-
buildsys@lists.rpmfusion.org>
Summary: gpg(RPM Fusion free repository for Fedora (28) <rpmfusion-
buildsys@lists.rpmfusion.org>)
Creation date: 2017-07-07
Key ID: 09eab3f2
```

The format of the package (`gpg-pubkey-9db62fb1-59920156`) is important to understand for verification. Using the above example, it consists of three parts:

1. The general prefix name for all imported GPG keys: `gpg-pubkey-`
2. The version, which is the GPG key ID: `9db62fb1`
3. The release is the date of the key in UNIX timestamp in hexadecimal: `59920156`

With both the date and the GPG key ID, check the relevant repositories public key page to confirm that the keys are indeed correct.

Query locally available GPG keys

Repositories that store their respective GPG keys on disk should do so in `/etc/pki/rpm-gpg/`. These keys are available for immediate import either when `dnf` is asked to install a relevant package from the repository or when an administrator imports the key directly with the `rpm --import` command.

To find where these keys come from run:

```
# for PACKAGE in $(find /etc/pki/rpm-gpg/ -type f -exec rpm -qf {} \; | sort
-u); do rpm -q --queryformat "%{NAME}-%{VERSION} %{PACKAGER} %{SUMMARY}\\n"
"${PACKAGE}"; done
```

Remediation:

Update your package manager GPG keys in accordance with site policy.

References:

1. NIST SP 800-53 Rev. 5: SI-2

Additional Information:

Fedora public keys: https://getfedora.org/security/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7.3 Perform Automated Operating System Patch Management Perform operating system updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v8	7.4 Perform Automated Application Patch Management Perform application updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v7	3.4 Deploy Automated Operating System Patch Management Tools Deploy automated software update tools in order to ensure that the operating systems are running the most recent security updates provided by the software vendor.	●	●	●
v7	3.5 Deploy Automated Software Patch Management Tools Deploy automated software update tools in order to ensure that third-party software on all systems is running the most recent security updates provided by the software vendor.	●	●	●

1.2.1.2 Ensure gpgcheck is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `gpgcheck` option controls verifying package signatures after download. This option is configurable as a global option in the main section of `/etc/dnf/dnf.conf` and a per repository option in individual files in the `/etc/yum.repos.d/*` directory.

The option is enabled if `gpgcheck` is set to `1`, `true`, or `yes`. The option is disabled if `gpgcheck` is set to `0`, `false`, or `no`. If an invalid option is set, e.g. `gpgcheck=2`, the global option will be used.

Settings in files in the `/etc/yum.repos.d/` directory take precedence over the global configuration.

Rationale:

It is important to ensure that an RPM's package signature is always checked prior to installation to ensure that the software is obtained from a trusted source.

Audit:

Run the following command and verify that global configuration for `gpgcheck` is enabled:

```
# grep -Pi -- '^\h*gpgcheck\h*=\h*(1|true|yes)\b' /etc/dnf/dnf.conf
```

Verify the output is: `gpgcheck=1`, `gpgcheck=true`, or `gpgcheck=yes`.
Example output:

```
gpgcheck=1
```

Run the following command to verify `gpgcheck` is not disabled in a file in the `/etc/yum.repos.d/` directory:

```
# grep -Pris -- '^\h*gpgcheck\h*=\h*(0|[2-9]|[1-9][0-9]+|false|no)\b'
/etc/yum.repos.d/
```

Nothing should be returned.

Remediation:

Edit `/etc/dnf/dnf.conf` and set `gpgcheck=1`:
Example

```
# sed -i 's/^gpgcheck\s*=\s*.*/gpgcheck=1/' /etc/dnf/dnf.conf
```

Edit any failing files in `/etc/yum.repos.d/*` and set all instances starting with `gpgcheck` to `1`.
Example:

```
# find /etc/yum.repos.d/ -name "*.repo" -exec echo "Checking:" {} \; -exec
sed -i 's/^gpgcheck\s*=\s*.*/gpgcheck=1/' {} \;
```

Default Value:

gpgcheck=1

References:

1. NIST SP 800-53 Revision 5 :: CM-14
2. STIG ID: RHEL-08-010370 | RULE ID: SV-230264r1017377 | CAT I
3. STIG ID: RHEL-09-214015 | RULE ID: SV-257820r1044878 | CAT I
4. STIG ID: RHEL-09-214025 | RULE ID: SV-257822r1044880 | CAT I

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7.3 Perform Automated Operating System Patch Management Perform operating system updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v7	3.4 Deploy Automated Operating System Patch Management Tools Deploy automated software update tools in order to ensure that the operating systems are running the most recent security updates provided by the software vendor.	●	●	●

1.2.1.3 Ensure TDNF gpgcheck is globally activated (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `gpgcheck` option, found in the main section of the `/etc/tdnf/tdnf.conf` and individual `/etc/yum.repos.d/*` files, determines if an RPM package's signature is checked prior to its installation.

Rationale:

It is important to ensure that an RPM's package signature is always checked prior to installation to ensure that the software is obtained from a trusted source.

Audit:

Global configuration. Run the following command and verify that `gpgcheck` is set to `1`:

```
# grep ^gpgcheck /etc/tdnf/tdnf.conf

gpgcheck=1
```

Configuration in `/etc/yum.repos.d/` takes precedence over the global configuration. Run the following command and verify that there are no instances of entries starting with `gpgcheck` returned set to `0`. Nor should there be any invalid (non-boolean) values. When `dnf` encounters such invalid entries they are ignored and the global configuration is applied.

```
# grep -P "^gpgcheck\h*=\h*[^1\n\r]\b" /etc/yum.repos.d/*
```

Remediation:

Edit `/etc/dnf/dnf.conf` and set `gpgcheck=1` in the `[main]` section.
Example:

```
# sed -i 's/^gpgcheck\s*=\s*.*/gpgcheck=1/' /etc/tdnf/tdnf.conf
```

Edit any failing files in `/etc/yum.repos.d/*` and set all instances starting with `gpgcheck` to `1`.
Example:

```
# find /etc/yum.repos.d/ -name "*.repo" -exec echo "Checking:" {} \; -exec
sed -i 's/^gpgcheck\s*=\s*.*/gpgcheck=1/' {} \;
```

References:

1. NIST SP 800-53 Rev. 5: - SI-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7.3 Perform Automated Operating System Patch Management Perform operating system updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v7	3.4 Deploy Automated Operating System Patch Management Tools Deploy automated software update tools in order to ensure that the operating systems are running the most recent security updates provided by the software vendor.	●	●	●

1.2.1.4 Ensure package manager repositories are configured (Manual)

Profile Applicability:

- Level 1 - Server

Description:

Systems need to have the respective package manager repositories configured to ensure that the system is able to receive the latest patches and updates.

Rationale:

If a system's package repositories are misconfigured, important patches may not be identified or a rogue repository could introduce compromised software.

Audit:

Run the following command to verify repositories are configured correctly. The output may vary depending on which repositories are currently configured on the system.
Example:

```
# dnf repolist
Last metadata expiration check: 1:00:00 ago on Mon 1 Jan 2021 00:00:00 BST.
repo id         repo name                          status
*fedora         Fedora 28 - x86_64                 57,327
*updates        Fedora 28 - x86_64 - Updates       22,133
```

For the repositories in use, inspect the configuration file to ensure all settings are correctly applied according to site policy.
Example:
Depending on the distribution being used the repo file name might differ.

```
cat /etc/yum.repos.d/*.repo
```

Remediation:

Configure your package manager repositories according to site policy.

References:

1. NIST SP 800-53 Rev. 5: SI-2

Additional Information:

For further information about Fedora repositories see: https://docs.fedoraproject.org/en-US/quick-docs/repositories/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7.3 Perform Automated Operating System Patch Management Perform operating system updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v8	7.4 Perform Automated Application Patch Management Perform application updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v7	3.4 Deploy Automated Operating System Patch Management Tools Deploy automated software update tools in order to ensure that the operating systems are running the most recent security updates provided by the software vendor.	●	●	●
v7	3.5 Deploy Automated Software Patch Management Tools Deploy automated software update tools in order to ensure that third-party software on all systems is running the most recent security updates provided by the software vendor.	●	●	●

1.3 Configure Additional Process Hardening

1.3.1 Ensure address space layout randomization is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Address space layout randomization (ASLR) is an exploit mitigation technique which randomly arranges the address space of key data areas of a process.

Rationale:

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

Audit:

Run the following script to verify the following kernel parameter is set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `kernel.randomize_va_space` is set to `2`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); a_parlist=(kernel.randomize_va_space=2)
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
```

```
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      f_kernel_parameter_chk
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameter in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `kernel.randomize_va_space = 2`

Example:

```
# printf '%s\n' "" "kernel.randomize_va_space = 2" >> /etc/sysctl.d/60-
kernel_sysctl.conf
```

Run the following command to set the active kernel parameter:

```
# sysctl -w kernel.randomize_va_space=2
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

kernel.randomize_va_space = 2

References:

1. CCI-000366: The organization implements the security configuration settings
2. NIST SP 800-53 Rev. 5: CM-6
3. NIST SP 800-53A :: CM-6.1 (iv)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	10.5 Enable Anti-Exploitation Features Enable anti-exploitation features on enterprise assets and software, where possible, such as Microsoft® Data Execution Prevention (DEP), Windows® Defender Exploit Guard (WDEG), or Apple® System Integrity Protection (SIP) and Gatekeeper™.		●	●
v7	8.3 Enable Operating System Anti-Exploitation Features/ Deploy Anti-Exploit Technologies Enable anti-exploitation features such as Data Execution Prevention (DEP) or Address Space Layout Randomization (ASLR) that are available in an operating system or deploy appropriate toolkits that can be configured to apply protection to a broader set of applications and executables.		●	●

1.3.2 Ensure ptrace_scope is restricted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `ptrace()` system call provides a means by which one process (the "tracer") may observe and control the execution of another process (the "tracee"), and examine and change the tracee's memory and registers.

The sysctl settings (writable only with CAP_SYS_PTRACE) are:

- `0` - classic ptrace permissions: a process can PTRACE_ATTACH to any other process running under the same uid, as long as it is dumpable (i.e. did not transition uids, start privileged, or have called prctl(PR_SET_DUMPABLE...) already). Similarly, PTRACE_TRACEME is unchanged.
- `1` - restricted ptrace: a process must have a predefined relationship with the inferior it wants to call PTRACE_ATTACH on. By default, this relationship is that of only its descendants when the above classic criteria is also met. To change the relationship, an inferior can call prctl(PR_SET_PTRACER, debugger, ...) to declare an allowed debugger PID to call PTRACE_ATTACH on the inferior. Using PTRACE_TRACEME is unchanged.
- `2` - admin-only attach: only processes with CAP_SYS_PTRACE may use ptrace with PTRACE_ATTACH, or through children calling PTRACE_TRACEME.
- `3` - no attach: no processes may use ptrace with PTRACE_ATTACH nor via PTRACE_TRACEME. Once set, this sysctl value cannot be changed.

Rationale:

If one application is compromised, it would be possible for an attacker to attach to other running processes (e.g. Bash, Firefox, SSH sessions, GPG agent, etc) to extract additional credentials and continue to expand the scope of their attack.

Enabling restricted mode will limit the ability of a compromised process to PTRACE_ATTACH on other processes running under the same user. With restricted mode, ptrace will continue to work with root user.

Audit:

Run the following script to verify the following kernel parameter is set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `kernel.yama.ptrace_scope` is set to a value of: `1`, `2`, or `3`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); a_parlist=("kernel.yama.ptrace_scope=(1|2|3)")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
```

```
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      f_kernel_parameter_chk
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the `kernel.yama.ptrace_scope` parameter in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf` to a value of `1`, `2`, or `3`:

```
kernel.yama.ptrace_scope = 1
    - OR -
kernel.yama.ptrace_scope = 2
    - OR -
kernel.yama.ptrace_scope = 3
```

Example:

```
# printf "%s\n" "kernel.yama.ptrace_scope = 1" >> /etc/sysctl.d/60-
kernel_sysctl.conf
```

Run the following command to set the active kernel parameter:

```
# sysctl -w kernel.yama.ptrace_scope=1
```

Note:

- If a value of `2` or `3` is preferred, or required by local site policy, replace the `1` with the desired value of `2` or `3` in the example above
- If this setting appears in a canonically later file, or later in the same file, the setting will be overwritten

Default Value:

kernel.yama.ptrace_scope = 0

References:

1. https://www.kernel.org/doc/Documentation/security/Yama.txt
2. https://github.com/raj3shp/termspy
3. NIST SP 800-53 Rev. 5: CM-6

Additional Information:

Ptrace is very rarely used by regular applications and is mostly used by debuggers such as `gdb` and `strace`.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

1.3.3 Ensure core dump backtraces are disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

A core dump is the memory of an executable program. It is generally used to determine why a program aborted. It can also be used to glean confidential information from a core file.

Rationale:

A core dump includes a memory image taken at the time the operating system terminates an application. The memory image could contain sensitive data and is generally useful only for developers trying to debug problems, increasing the risk to the system.

Audit:

Run the following script to verify `ProcessSizeMax` is set to `0` in `/etc/systemd/coredump.conf` or a file in the `/etc/systemd/coredump.conf.d/` directory:

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); a_parlist=("ProcessSizeMax=0")
   l_systemd_config_file="/etc/systemd/coredump.conf" # Main systemd
configuration file
   f_config_file_parameter_chk()
   {
      unset A_out; declare -A A_out # Check config file(s) setting
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_systemd_parameter="$(awk -F= '{print $1}' <<< "$l_out" |
xargs)"
               grep -Piq -- "^\h*$l_systemd_parameter_name\b" <<<
"$l_systemd_parameter" && A_out+=(["$l_systemd_parameter"]="$l_file")
            fi
         fi
      done < <("$l_systemdanalyze" cat-config "$l_systemd_config_file" | grep
-Pio '^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_systemd_file_parameter_name
l_systemd_file_parameter_value; do
            l_systemd_file_parameter_name="${l_systemd_file_parameter_name//
/}"

l_systemd_file_parameter_value="${l_systemd_file_parameter_value// /}"
            if grep -Piq "\b$l_systemd_parameter_value\b" <<<
"$l_systemd_file_parameter_value"; then
               a_output+=(" - \"$l_systemd_parameter_name\" is correctly set
to \"$l_systemd_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_systemd_parameter_name\" is incorrectly
set to \"$l_systemd_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\" and should have a
value matching: \"$l_value_out\"")
            fi
         done < <(grep -Pio -- "^\h*$l_systemd_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2=(" - \"$l_systemd_parameter_name\" is not set in an
included file" \
         "   *** Note: \"$l_systemd_parameter_name\" May be set in a file
that's ignored by load procedure ***")
      fi
   }
   l_systemdanalyze="$(readlink -f /bin/systemd-analyze)"
   while IFS="=" read -r l_systemd_parameter_name l_systemd_parameter_value;
do # Assess and check parameters
      l_systemd_parameter_name="${l_systemd_parameter_name// /}";
l_systemd_parameter_value="${l_systemd_parameter_value// /}"
      l_value_out="${l_systemd_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
```

```
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      f_config_file_parameter_chk
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Create or edit the file `/etc/systemd/coredump.conf`, or a file in the `/etc/systemd/coredump.conf.d` directory ending in `.conf`.
Edit or add the following line in the `[Coredump]` section:

```
ProcessSizeMax=0
```

Example:

```
#!/usr/bin/env bash

{
   [ ! -d /etc/systemd/coredump.conf.d/ ] && mkdir
/etc/systemd/coredump.conf.d/
   if grep -Psq -- '^\h*\[Coredump\]' /etc/systemd/coredump.conf.d/60-
coredump.conf; then
      printf '%s\n' "ProcessSizeMax=0" >> /etc/systemd/coredump.conf.d/60-
coredump.conf
   else
      printf '%s\n' "[Coredump]" "ProcessSizeMax=0" >>
/etc/systemd/coredump.conf.d/60-coredump.conf
   fi
}
```

Default Value:

ProcessSizeMax=2G

References:

1. https://www.freedesktop.org/software/systemd/man/coredump.conf.html
2. NIST SP 800-53 Rev. 5: CM-6b

1.3.4 Ensure core dump storage is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

A core dump is the memory of an executable program. It is generally used to determine why a program aborted. It can also be used to glean confidential information from a core file.

Rationale:

A core dump includes a memory image taken at the time the operating system terminates an application. The memory image could contain sensitive data and is generally useful only for developers trying to debug problems.

Audit:

Run the following script to verify `Storage` is set to `none` in `/etc/systemd/coredump.conf` or a file in the `/etc/systemd/coredump.conf.d/` directory:

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); a_parlist=("Storage=none")
   l_systemd_config_file="/etc/systemd/coredump.conf" # Main systemd
configuration file
   f_config_file_parameter_chk()
   {
      unset A_out; declare -A A_out # Check config file(s) setting
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_systemd_parameter="$(awk -F= '{print $1}' <<< "$l_out" |
xargs)"
               grep -Piq -- "^\h*$l_systemd_parameter_name\b" <<<
"$l_systemd_parameter" && A_out+=(["$l_systemd_parameter"]="$l_file")
            fi
         fi
      done < <("$l_systemdanalyze" cat-config "$l_systemd_config_file" | grep
-Pio '^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_systemd_file_parameter_name
l_systemd_file_parameter_value; do
            l_systemd_file_parameter_name="${l_systemd_file_parameter_name//
/}"

l_systemd_file_parameter_value="${l_systemd_file_parameter_value// /}"
            if grep -Piq "\b$l_systemd_parameter_value\b" <<<
"$l_systemd_file_parameter_value"; then
               a_output+=(" - \"$l_systemd_parameter_name\" is correctly set
to \"$l_systemd_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_systemd_parameter_name\" is incorrectly
set to \"$l_systemd_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\" and should have a
value matching: \"$l_value_out\"")
            fi
         done < <(grep -Pio -- "^\h*$l_systemd_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2=(" - \"$l_systemd_parameter_name\" is not set in an
included file" \
         "   *** Note: \"$l_systemd_parameter_name\" May be set in a file
that's ignored by load procedure ***")
      fi
   }
   l_systemdanalyze="$(readlink -f /bin/systemd-analyze)"
   while IFS="=" read -r l_systemd_parameter_name l_systemd_parameter_value;
do # Assess and check parameters
      l_systemd_parameter_name="${l_systemd_parameter_name// /}";
l_systemd_parameter_value="${l_systemd_parameter_value// /}"
      l_value_out="${l_systemd_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
```

```
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      f_config_file_parameter_chk
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Create or edit the file `/etc/systemd/coredump.conf`, or a file in the `/etc/systemd/coredump.conf.d` directory ending in `.conf`.
Edit or add the following line in the `[Coredump]` section:

```
Storage=none
```

Example:

```
#!/usr/bin/env bash

{
   [ ! -d /etc/systemd/coredump.conf.d/ ] && mkdir
/etc/systemd/coredump.conf.d/
   if grep -Psq -- '^\h*\[Coredump\]' /etc/systemd/coredump.conf.d/60-
coredump.conf; then
      printf '%s\n' "Storage=none" >> /etc/systemd/coredump.conf.d/60-
coredump.conf
   else
      printf '%s\n' "[Coredump]" "Storage=none" >>
/etc/systemd/coredump.conf.d/60-coredump.conf
   fi
}
```

References:

1. https://www.freedesktop.org/software/systemd/man/coredump.conf.html

1.4 Configure Command Line Warning Banners

Presenting a warning message prior to the normal user login may assist in the prosecution of trespassers on the computer system. Changing some of these login banners also has the side effect of hiding OS version information and other detailed system information from attackers attempting to target specific exploits at a system.

Guidelines published by the US Department of Defense require that warning messages include at least the name of the organization that owns the system, the fact that the system is subject to monitoring and that such monitoring is in compliance with local statutes, and that use of the system implies consent to such monitoring. It is important that the organization's legal counsel review the content of all messages before any system modifications are made, as these warning messages are inherently site-specific. More information (including citations of relevant case law) can be found at http://www.justice.gov/criminal/cybercrime/

The `/etc/motd`, `/etc/issue`, and `/etc/issue.net` files govern warning banners for standard command line logins for both local and remote users.

Note: The text provided in the remediation actions for these items is intended as an example only. Please edit to include the specific text for your organization as approved by your legal department.

1.4.1 Ensure local login warning banner is configured properly (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The contents of the `/etc/issue` file are displayed to users prior to login for local terminals.

Unix-based systems have typically displayed information about the OS release and patch level upon logging in to the system. This information can be useful to developers who are developing software for a particular OS platform. If `mingetty(8)` supports the following options, they display operating system information: `\m` - machine architecture `\r` - operating system release `\s` - operating system name `\v` - operating system version - or the operating system's name.

Rationale:

Warning messages inform users who are attempting to login to the system of their legal status regarding the system and must include the name of the organization that owns the system and any monitoring policies that are in place. Displaying OS and patch level information in login banners also has the side effect of providing detailed system information to attackers attempting to target specific exploits of a system. Authorized users can easily get this information by running the " `uname -a` " command once they have logged in.

Audit:

Run the following command and verify that the contents match site policy:

```
# cat /etc/issue
```

Run the following command and verify no results are returned:

```
# grep -E -i "(\\v|\\r|\\m|\\s|$(grep '^ID=' /etc/os-release | cut -d= -
f2 | sed -e 's/"//g'))" /etc/issue
```

Remediation:

Edit the `/etc/issue` file with the appropriate contents according to your site policy, remove any instances of `\m` , `\r` , `\s` , `\v` or references to the `OS platform`.
Example:

```
# echo "Authorized users only. All activity may be monitored and reported." >
/etc/issue
```

References:

1. NIST SP 800-53 Rev. 5: CM-6, CM-1, CM-3

1.4.2 Ensure remote login warning banner is configured properly (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The contents of the `/etc/issue.net` file are displayed to users prior to login for remote connections from configured services.

Unix-based systems have typically displayed information about the OS release and patch level upon logging in to the system. This information can be useful to developers who are developing software for a particular OS platform. If `mingetty(8)` supports the following options, they display operating system information: `\m` - machine architecture `\r` - operating system release `\s` - operating system name `\v` - operating system version.

Rationale:

Warning messages inform users who are attempting to login to the system of their legal status regarding the system and must include the name of the organization that owns the system and any monitoring policies that are in place. Displaying OS and patch level information in login banners also has the side effect of providing detailed system information to attackers attempting to target specific exploits of a system. Authorized users can easily get this information by running the " `uname -a` " command once they have logged in.

Audit:

Run the following command and verify that the contents match site policy:

```
# cat /etc/issue.net
```

Run the following command and verify no results are returned:

```
# grep -E -i "(\\v|\\r|\\m|\\s|$(grep '^ID=' /etc/os-release | cut -d= -
f2 | sed -e 's/"//g'))" /etc/issue.net
```

Remediation:

Edit the `/etc/issue.net` file with the appropriate contents according to your site policy, remove any instances of `\m` , `\r` , `\s` , `\v` or references to the `OS platform`.
Example:

```
# echo "Authorized users only. All activity may be monitored and reported." >
/etc/issue.net
```

References:

1. NIST SP 800-53 Rev. 5: CM-6, CM-1, CM-3

1.4.3 Ensure access to /etc/motd is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The contents of the `/etc/motd` file are displayed to users after login and function as a message of the day for authenticated users.

Rationale:

- IF - the `/etc/motd` file does not have the correct access configured, it could be modified by unauthorized users with incorrect or misleading information.

Audit:

Run the following command and verify that if `/etc/motd` exists, `Access` is `644` or more restrictive, `Uid` and `Gid` are both `0/root`:

```
# [ -e /etc/motd ] && stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: { %g/
%G)' /etc/motd

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: ( 0/ root)
  -- OR --
Nothing is returned
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/motd`:

```
# chown root:root $(readlink -e /etc/motd)
# chmod u-x,go-wx $(readlink -e /etc/motd)
```

- OR -
Run the following command to remove the `/etc/motd` file:

```
# rm /etc/motd
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

1.4.4 Ensure access to /etc/issue is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The contents of the `/etc/issue` file are displayed to users prior to login for local terminals.

Rationale:

- IF - the `/etc/issue` file does not have the correct access configured, it could be modified by unauthorized users with incorrect or misleading information.

Audit:

Run the following command and verify `Access` is `644` or more restrictive and `Uid` and `Gid` are both `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: { %g/ %G)' /etc/issue

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: { 0/ root)
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/issue`:

```
# chown root:root $(readlink -e /etc/issue)
# chmod u-x,go-wx $(readlink -e /etc/issue)
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

1.4.5 Ensure access to /etc/issue.net is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The contents of the `/etc/issue.net` file are displayed to users prior to login for remote connections from configured services.

Rationale:

- IF - the `/etc/issue.net` file does not have the correct access configured, it could be modified by unauthorized users with incorrect or misleading information.

Audit:

Run the following command and verify `Access` is `644` or more restrictive and `Uid` and `Gid` are both `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: { %g/ %G)' /etc/issue.net

Access: (0644/-rw-r--r--)  Uid: ( 0/ root)   Gid: ( 0/ root)
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/issue.net`:

```
# chown root:root $(readlink -e /etc/issue.net)
# chmod u-x,go-wx $(readlink -e /etc/issue.net)
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

2 Services

While applying system updates and patches helps correct known vulnerabilities, one of the best ways to protect the system against as yet unreported vulnerabilities is to disable all services that are not required for normal system operation. This prevents the exploitation of vulnerabilities discovered at a later date. If a service is not enabled, it cannot be exploited. The actions in this section of the document provide guidance on some services which can be safely disabled and under which circumstances, greatly reducing the number of possible threats to the resulting system. Additionally some services which should remain enabled but with secure configuration are covered as well as insecure service clients.

2.1 Configure Time Synchronization

It is recommended that physical systems and virtual guests lacking direct access to the physical host's clock be configured to synchronize their time using a service such as NTP or chrony.

2.1.1 Ensure time synchronization is in use (Automated)

Profile Applicability:

- Level 1 - Server

Description:

System time should be synchronized between all systems in an environment. This is typically done by establishing an authoritative time server or set of servers and having all systems synchronize their clocks to them.

Note: If another method for time synchronization is being used, this section may be skipped.

Rationale:

Time synchronization is important to support time sensitive security mechanisms like Kerberos and also ensures log files have consistent time records across the enterprise, which aids in forensic investigations.

Audit:

Run the following commands to verify that chrony is installed:

```
# rpm -q chrony

chrony-<version>
```

Remediation:

Run the following command to install `chrony`:

```
# tdnf install chrony
```

References:

1. NIST SP 800-53 Rev. 5: AU-3, AU-12

Additional Information:

On systems where host based time synchronization is not available, verify that chrony is installed.

On systems where host based time synchronization is available consult your documentation and verify that host based synchronization is in use.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.4 Standardize Time Synchronization Standardize time synchronization. Configure at least two synchronized time sources across enterprise assets, where supported.		●	●
v7	6.1 Utilize Three Synchronized Time Sources Use at least three synchronized time sources from which all servers and network devices retrieve time information on a regular basis so that timestamps in logs are consistent.		●	●

2.1.2 Ensure chrony is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`chrony` is a daemon which implements the Network Time Protocol (NTP) and is designed to synchronize system clocks across a variety of systems and use a source that is highly accurate. More information on `chrony` can be found at http://chrony.tuxfamily.org/. `chrony` can be configured to be a client and/or a server.

Rationale:

If chrony is in use on the system proper configuration is vital to ensuring time synchronization is working properly.

Audit:

Run the following command and verify remote server is configured properly:

```
# grep -E "^(server|pool|refclock)" /etc/chrony.conf

server <remote-server>
```

Multiple servers may be configured.
Run the following command and verify `OPTIONS` includes '`-u chrony`':

```
# grep ^OPTIONS /etc/sysconfig/chronyd

OPTIONS="-u chrony"
```

Additional options may be present.

Remediation:

Add or edit server or pool lines to `/etc/chrony.conf` as appropriate:

```
server <remote-server>
```

Add or edit the `OPTIONS` in `/etc/sysconfig/chronyd` to include '`-u chrony`':

```
OPTIONS="-u chrony"
```

References:

1. NIST SP 800-53 Rev. 5: AU-3, AU-12

Additional Information:

On systems where host based time synchronization is not available, verify that chrony is installed.

On systems where host based time synchronization is available consult your documentation and verify that host based synchronization is in use.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.4 Standardize Time Synchronization Standardize time synchronization. Configure at least two synchronized time sources across enterprise assets, where supported.		●	●
v7	6.1 Utilize Three Synchronized Time Sources Use at least three synchronized time sources from which all servers and network devices retrieve time information on a regular basis so that timestamps in logs are consistent.		●	●

2.2 Configure Special Purpose Services

This section describes services that are installed on systems that specifically need to run these services. If any of these services are not required, it is recommended that the package be removed, or the service be masked to reduce the potential attack surface.

Note: This should not be considered a comprehensive list of services not required for normal system operation. You may wish to consider additions to those listed here for your environment

2.2.1 Ensure xinetd is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The eXtended InterNET Daemon (`xinetd`) is an open source super daemon that replaced the original `inetd` daemon. The `xinetd` daemon listens for well known services and dispatches the appropriate daemon to properly respond to service requests.

Rationale:

If there are no xinetd services required, it is recommended that the package be removed to reduce the attack surface area of the system.

Note: If an xinetd service or services are required, ensure that any xinetd service not required is stopped and disabled.

Audit:

Run the following command to verify `xinetd` is not installed:

```
# rpm -q xinetd

package xinetd is not installed
```

Remediation:

Run the following command to remove `xinetd`:

```
# tdnf remove xinetd
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.2 Ensure xorg-x11-server-common is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The X Window System provides a Graphical User Interface (GUI) where users can have multiple windows in which to run programs and various add on. The X Windows system is typically used on workstations where users login, but not on servers where users typically do not login.

Rationale:

Unless your organization specifically requires graphical login access via X Windows, remove it to reduce the potential attack surface.

Impact:

Many Linux systems run applications which require a Java runtime. Some Linux Java packages have a dependency on specific X Windows xorg-x11-fonts. One workaround to avoid this dependency is to use the "headless" Java packages for your specific Java runtime.

Audit:

Run the following command to verify X Windows Server is not installed.

```
# rpm -q xorg-x11-server-common

package xorg-x11-server-common is not installed
```

Remediation:

Run the following command to remove the X Windows Server packages:

```
# tdnf remove xorg-x11-server-common
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.3 Ensure avahi is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Avahi is a free zeroconf implementation, including a system for multicast DNS/DNS-SD service discovery. Avahi allows programs to publish and discover services and hosts running on a local network with no specific configuration. For example, a user can plug a computer into a network and Avahi automatically finds printers to print to, files to look at and people to talk to, as well as network services running on the machine.

Rationale:

Automatic discovery of network services is not normally required for system functionality. It is recommended to remove this package to reduce the potential attack surface.

Audit:

Run the following command to verify `avahi` is not installed:

```
# rpm -q avahi

package avahi is not installed
```

Remediation:

Run the following commands to stop, and remove `avahi`:

```
# systemctl stop avahi-daemon.socket avahi-daemon.service
# tdnf remove avahi
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.4 Ensure a print server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Common Unix Print System (CUPS) provides the ability to print to both local and network printers. A system running CUPS can also accept print jobs from remote systems and print them to local printers. It also provides a web based remote administration capability.

Rationale:

If the system does not need to print jobs or accept print jobs from other systems, it is recommended that CUPS be removed to reduce the potential attack surface.

Note: Removing CUPS will prevent printing from the system

Impact:

Disabling CUPS will prevent printing from the system, a common task for workstation systems.

Audit:

Run the following command to verify `cups` is not installed:

```
# rpm -q cups

package cups is not installed
```

Remediation:

Run the following command to remove `cups`:

```
# tdnf remove cups
```

References:

1. More detailed documentation on CUPS is available at the project homepage at http://www.cups.org.
2. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.5 Ensure a dhcp server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Dynamic Host Configuration Protocol (DHCP) is a service that allows machines to be dynamically assigned IP addresses.

Rationale:

Unless a system is specifically set up to act as a DHCP server, it is recommended that the `dhcp-server` package be removed to reduce the potential attack surface.

Audit:

Run the following command to verify `dhcp-server` is not installed:

```
# rpm -q dhcp-server

package dhcp-server is not installed
```

Remediation:

Run the following command to remove `dhcp`:

```
# tdnf remove dhcp-server
```

References:

1. dhcpd(8)

Additional Information:

NIST SP 800-53 Rev. 5:

- CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.6 Ensure a dns server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Domain Name System (DNS) is a hierarchical naming system that maps names to IP addresses for computers, services and other resources connected to a network.

Rationale:

Unless a system is specifically designated to act as a DNS server, it is recommended that the package be removed to reduce the potential attack surface.

Audit:

Run one of the following commands to verify `bind` is not installed:

```
# rpm -q bind

package bind is not installed
```

Remediation:

Run the following command to remove `bind`:

```
# tdnf remove bind
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.7 Ensure FTP client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

FTP (File Transfer Protocol) is a traditional and widely used standard tool for transferring files between a server and clients over a network, especially where no authentication is necessary (permits anonymous users to connect to a server).

Rationale:

FTP does not protect the confidentiality of data or authentication credentials. It is recommended SFTP be used if file transfer is required. Unless there is a need to run the system as a FTP server (for example, to allow anonymous downloads), it is recommended that the package be removed to reduce the potential attack surface.

Audit:

Run the following command to verify `ftp` is not installed:

```
# rpm -q ftp

package ftp is not installed
```

Remediation:

Run the following command to remove `ftp`:

```
# tdnf remove ftp
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.8 Ensure an ftp server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

FTP (File Transfer Protocol) is a traditional and widely used standard tool for transferring files between a server and clients over a network, especially where no authentication is necessary (permits anonymous users to connect to a server).

Rationale:

Unless there is a need to run the system as a FTP server, it is recommended that the package be removed to reduce the potential attack surface.

Audit:

Run the following command to verify `vsftpd` is not installed:

```
# rpm -q vsftpd

package vsftpd is not installed
```

Remediation:

Run the following command to remove `vsftpd`:

```
# tdnf remove vsftpd
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.9 Ensure a tftp server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Trivial File Transfer Protocol (TFTP) is a simple protocol for exchanging files between two TCP/IP machines. TFTP servers allow connections from a TFTP Client for sending and receiving files.

Rationale:

Unless there is a need to run the system as a TFTP server, it is recommended that the package be removed to reduce the potential attack surface.

TFTP does not have built-in encryption, access control or authentication. This makes it very easy for an attacker to exploit TFTP to gain access to files

Impact:

TFTP is often used to provide files for network booting such as for PXE based installation of servers.

Audit:

Run the following command to verify `tftp-server` is not installed:

```
# rpm -q tftp-server

package tftp-server is not installed
```

Remediation:

Run the following command to remove `tftp-server`:

```
# tdnf remove tftp-server
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.10 Ensure a web server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Web servers provide the ability to host web site content.

Rationale:

Unless there is a need to run the system as a web server, it is recommended that the packages be removed to reduce the potential attack surface.

Note: Several http servers exist. They should also be audited, and removed, if not required.

Audit:

Run the following command to verify `httpd` and `nginx` are not installed:

```
# rpm -q httpd nginx

package httpd is not installed
package nginx is not installed
```

Remediation:

Run the following command to remove `httpd` and `nginx`:

```
# tdnf remove httpd nginx
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.11 Ensure IMAP and POP3 server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`dovecot` is an open source IMAP and POP3 server for Linux based systems.

Rationale:

Unless POP3 and/or IMAP servers are to be provided by this system, it is recommended that the package be removed to reduce the potential attack surface.

Note: Several IMAP/POP3 servers exist and can use other service names. These should also be audited and the packages removed if not required.

Audit:

Run the following command to verify `dovecot` and `cyrus-imapd` are not installed:

```
# rpm -q dovecot cyrus-imapd

package dovecot is not installed
package cyrus-imapd is not installed
```

Remediation:

Run the following command to remove `dovecot` and `cyrus-imapd`:

```
# tdnf remove dovecot cyrus-imapd
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.12 Ensure Samba is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Samba daemon allows system administrators to configure their Linux systems to share file systems and directories with Windows desktops. Samba will advertise the file systems and directories via the Server Message Block (SMB) protocol. Windows desktop users will be able to mount these directories and file systems as letter drives on their systems.

Rationale:

If there is no need to mount directories and file systems to Windows systems, then this package can be removed to reduce the potential attack surface.

Audit:

Run the following command to verify `samba` is not installed:

```
# rpm -q samba

package samba is not installed
```

Remediation:

Run the following command to remove `samba`:

```
# tdnf remove samba
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.13 Ensure HTTP Proxy Server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Squid is a standard proxy server used in many distributions and environments.

Rationale:

Unless a system is specifically set up to act as a proxy server, it is recommended that the squid package be removed to reduce the potential attack surface.

Note: Several HTTP proxy servers exist. These should be checked and removed unless required.

Audit:

Run the following command to verify `squid` is not installed:

```
# rpm -q squid

package squid is not installed
```

Remediation:

Run the following command to remove the `squid` package:

```
# tdnf remove squid
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.14 Ensure net-snmp is not installed or the snmpd service is not enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Simple Network Management Protocol (SNMP) is a widely used protocol for monitoring the health and welfare of network equipment, computer equipment and devices like UPSs.

Net-SNMP is a suite of applications used to implement SNMPv1 (RFC 1157), SNMPv2 (RFCs 1901-1908), and SNMPv3 (RFCs 3411-3418) using both IPv4 and IPv6.

Support for SNMPv2 classic (a.k.a. "SNMPv2 historic" - RFCs 1441-1452) was dropped with the 4.0 release of the UCD-snmp package.

The Simple Network Management Protocol (SNMP) server is used to listen for SNMP commands from an SNMP management system, execute the commands or collect the information and then send results back to the requesting system.

Rationale:

The SNMP server can communicate using `SNMPv1`, which transmits data in the clear and does not require authentication to execute commands. `SNMPv3` replaces the simple/clear text password sharing used in `SNMPv2` with more securely encoded parameters. If the the SNMP service is not required, the `net-snmp` package should be removed to reduce the attack surface of the system.

Note: If a required dependency exists for the net-snmp package, but the snmpd service is not required, the service should be masked.

Note: If SNMP is required:

- The server should be configured for `SNMP v3` only. `User Authentication` and `Message Encryption` should be configured.
- If `SNMP v2` is **absolutely** necessary, modify the community strings' values.

Impact:

There are packages that are dependent on the net-snmp package. If the net-snmp package is removed, these packages will be removed as well.

Before removing the net-snmp package, review any dependent packages to determine if they are required on the system. If a dependent package is required, mask the snmpd service and leave the net-snmp package installed.

Audit:

Run the following command to verify `net-snmp` is not installed:

```
# rpm -q net-snmp

package net-snmp is not installed
```

-OR-
Run the following command to verify the `snmpd` service is not enabled:

```
# systemctl is-enabled snmpd

masked
```

Verify output is not `enabled`.

Remediation:

Run the following command to remove `net-snmpd`:

```
# tdnf remove net-snmp
```

-OR-
Run the following commands to stop and mask the `snmpd` service:

```
# systemctl stop snmpd
# systemctl mask snmpd
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.15 Ensure NIS server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `ypserv` package provides the Network Information Service (NIS). This service, formally known as Yellow Pages, is a client-server directory service protocol for distributing system configuration files. The NIS server is a collection of programs that allow for the distribution of configuration files.

Rationale:

The NIS service is inherently an insecure system that has been vulnerable to DOS attacks, buffer overflows and has poor authentication for querying NIS maps. NIS generally has been replaced by such protocols as Lightweight Directory Access Protocol (LDAP). It is recommended that the `ypserv` package be removed, and if required, a more secure service be used.

Audit:

Run the following command to verify `ypserv` is not installed:

```
# rpm -q ypserv

package ypserv is not installed
```

Remediation:

Run the following command to remove `ypserv`:

```
# tdnf remove ypserv
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.16 Ensure telnet-server is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `telnet-server` package contains the `telnet` daemon, which accepts connections from users from other systems via the `telnet` protocol.

Rationale:

The `telnet` protocol is insecure and unencrypted. The use of an unencrypted transmission medium could allow a user with access to sniff network traffic the ability to steal credentials. The `ssh` package provides an encrypted session and stronger security.

Audit:

Run the following command to verify the `telnet-server` package is not installed:

```
rpm -q telnet-server

package telnet-server is not installed
```

Remediation:

Run the following command to remove the telnet-server package:

```
# tdnf remove telnet-server
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.17 Ensure mail transfer agent is configured for local-only mode (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Mail Transfer Agents (MTA), such as sendmail and Postfix, are used to listen for incoming mail and transfer the messages to the appropriate user or mail server. If the system is not intended to be a mail server, it is recommended that the MTA be configured to only process local mail.

Rationale:

The software for all Mail Transfer Agents is complex and most have a long history of security issues. While it is important to ensure that the system can process local mail messages, it is not necessary to have the MTA's daemon listening on a port unless the server is intended to be a mail server that receives and processes mail from other systems.

Note:

- This recommendation is designed around the postfix mail server.
- Depending on your environment you may have an alternative MTA installed such as sendmail. If this is the case consult the documentation for your installed MTA to configure the recommended state.

Audit:

Run the following command to verify that the MTA is not listening on any non-loopback address (`127.0.0.1` or `::1`)
Nothing should be returned.

```
#  ss -lntu | grep -P ':25\b' | grep -Pv '\h+(127\.0\.0\.1|\[?::1\]?):25\b'
```

Remediation:

Edit `/etc/postfix/main.cf` and add the following line to the RECEIVING MAIL section. If the line already exists, change it to look like the line below:

```
inet_interfaces = loopback-only
```

Run the following command to restart `postfix`:

```
# systemctl restart postfix
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.18 Ensure nfs-utils is not installed or the nfs-server service is masked (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Network File System (NFS) is one of the first and most widely distributed file systems in the UNIX environment. It provides the ability for systems to mount file systems of other servers through the network.

Rationale:

If the system does not require network shares, it is recommended that the nfs-utils package be removed to reduce the attack surface of the system.

Impact:

Many of the `libvirt` packages used by Enterprise Linux virtualization are dependent on the `nfs-utils` package. If the `nfs-package` is required as a dependency, the `nfs-server` should be disabled and masked to reduce the attack surface of the system.

Audit:

Run the following command to verify `nfs-utils` is not installed:

```
# rpm -q nfs-utils

package nfs-utils is not installed
```

OR
If the nfs-package is required as a dependency, run the following command to verify that the `nfs-server` service is masked:

```
# systemctl is-enabled nfs-server

masked
```

Remediation:

Run the following command to remove `nfs-utils`:

```
# tdnf remove nfs-utils
```

OR
If the nfs-package is required as a dependency, run the following command to stop and mask the `nfs-server` service:

```
# systemctl --now mask nfs-server
```

References:

1. NIST SP 800-53 Rev. 5: CM-6, CM-7

Additional Information:

Many of the libvirt packages used by Enterprise Linux virtualization are dependent on the nfs-utils package. If the nfs-package is required as a dependency, the nfs-server should be disabled and masked to reduce the attack surface of the system.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.2.19 Ensure rsync-daemon is not installed or the rsyncd service is masked (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `rsyncd` service can be used to synchronize files between systems over network links.

Rationale:

Unless required, the `rsync-daemon` package should be removed to reduce the attack surface area of the system.

The `rsyncd` service presents a security risk as it uses unencrypted protocols for communication.

Note: If a required dependency exists for the `rsync-daemon` package, but the `rsyncd` service is not required, the service should be masked.

Impact:

There are packages that are dependent on the rsync package. If the rsync package is removed, these packages will be removed as well.

Before removing the rsync package, review any dependent packages to determine if they are required on the system. If a dependent package is required, mask the rsyncd service and leave the rsync package installed.

Audit:

Run the following command to verify that `rsync` is not installed:

```
# rpm -q rsync-daemon

package rsync is not installed
```

OR
Run the following command to verify the `rsyncd` service is masked:

```
# systemctl is-enabled rsyncd

masked
```

Remediation:

Run the following command to remove the `rsync` package:

```
# tdnf remove rsync-daemon
```

OR
Run the following command to mask the `rsyncd` service:

```
# systemctl --now mask rsyncd
```

References:

1. NIST SP 800-53 Rev. 5: CM-6, CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

2.3 Service Clients

A number of insecure services exist. While disabling the servers prevents a local attack against these services, it is advised to remove their clients unless they are required.

Note: This should not be considered a comprehensive list of insecure service clients. You may wish to consider additions to those listed here for your environment.

2.3.1 Ensure NIS Client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Network Information Service (NIS), formerly known as Yellow Pages, is a client-server directory service protocol used to distribute system configuration files. The NIS client (`ypbind`) was used to bind a machine to an NIS server and receive the distributed configuration files.

Rationale:

The NIS service is inherently an insecure system that has been vulnerable to DOS attacks, buffer overflows and has poor authentication for querying NIS maps. NIS generally has been replaced by such protocols as Lightweight Directory Access Protocol (LDAP). It is recommended that the service be removed.

Impact:

Many insecure service clients are used as troubleshooting tools and in testing environments. Uninstalling them can inhibit capability to test and troubleshoot. If they are required it is advisable to remove the clients after use to prevent accidental or intentional misuse.

Audit:

Run the following command to verify that the `ypbind` package is not installed:

```
# rpm -q ypbind

package ypbind is not installed
```

Remediation:

Run the following command to remove the `ypbind` package:

```
# tdnf remove ypbind
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●

2.3.2 Ensure rsh client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `rsh` package contains the client commands for the rsh services.

Rationale:

These legacy clients contain numerous security exposures and have been replaced with the more secure SSH package. Even if the server is removed, it is best to ensure the clients are also removed to prevent users from inadvertently attempting to use these commands and therefore exposing their credentials. Note that removing the `rsh` package removes the clients for `rsh` , `rcp` and `rlogin` .

Impact:

Many insecure service clients are used as troubleshooting tools and in testing environments. Uninstalling them can inhibit capability to test and troubleshoot. If they are required it is advisable to remove the clients after use to prevent accidental or intentional misuse.

Audit:

Run the following command to verify that the `rsh` package is not installed:

```
# rpm -q rsh

package rsh is not installed
```

Remediation:

Run the following command to remove the `rsh` package:

```
# tdnf remove rsh
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●

2.3.3 Ensure talk client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `talk` software makes it possible for users to send and receive messages across systems through a terminal session. The `talk` client, which allows initialization of talk sessions, is installed by default.

Rationale:

The software presents a security risk as it uses unencrypted protocols for communication.

Impact:

Many insecure service clients are used as troubleshooting tools and in testing environments. Uninstalling them can inhibit capability to test and troubleshoot. If they are required it is advisable to remove the clients after use to prevent accidental or intentional misuse.

Audit:

Run the following command to verify that the `talk` package is not installed:

```
# rpm -q talk

package talk is not installed
```

Remediation:

Run the following command to remove the `talk` package:

```
# tdnf remove talk
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●

2.3.4 Ensure telnet client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `telnet` package contains the `telnet` client, which allows users to start connections to other systems via the telnet protocol.

Rationale:

The `telnet` protocol is insecure and unencrypted. The use of an unencrypted transmission medium could allow an unauthorized user to steal credentials. The `ssh` package provides an encrypted session and stronger security and is included in most Linux distributions.

Impact:

Many insecure service clients are used as troubleshooting tools and in testing environments. Uninstalling them can inhibit capability to test and troubleshoot. If they are required it is advisable to remove the clients after use to prevent accidental or intentional misuse.

Audit:

Run the following command to verify that the `telnet` package is not installed:

```
# rpm -q telnet

package telnet is not installed
```

Remediation:

Run the following command to remove the `telnet` package:

```
# tdnf remove telnet
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●

2.3.5 Ensure LDAP client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Lightweight Directory Access Protocol (LDAP) was introduced as a replacement for NIS/YP. It is a service that provides a method for looking up information from a central database.

Rationale:

If the system will not need to act as an LDAP client, it is recommended that the software be removed to reduce the potential attack surface.

Impact:

Removing the LDAP client will prevent or inhibit using LDAP for authentication in your environment.

Audit:

Run the following command to verify that the `openldap-clients` package is not installed:

```
# rpm -q openldap-clients

package openldap-clients is not installed
```

Remediation:

Run the following command to remove the `openldap-clients` package:

```
# tdnf remove openldap-clients
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	2.6 Address unapproved software Ensure that unauthorized software is either removed or the inventory is updated in a timely manner	●	●	●

2.3.6 Ensure TFTP client is not installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Trivial File Transfer Protocol (TFTP) is a simple protocol for exchanging files between two TCP/IP machines. TFTP servers allow connections from a TFTP Client for sending and receiving files.

Rationale:

TFTP does not have built-in encryption, access control or authentication. This makes it very easy for an attacker to exploit TFTP to gain access to files.

Audit:

Run the following command to verify `tftp` is not installed:

```
# rpm -q tftp

package tftp is not installed
```

Remediation:

Run the following command to remove `tftp`:

```
# tdnf remove tftp
```

References:

1. NIST SP 800-53 Rev. 5: CM-7

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3 Network

This section provides guidance on for securing the network configuration of the system.

3.1 Configure Network Kernel Parameters

The following network parameters are intended for use on both host only and router systems. A system acts as a router if it has at least two interfaces and is configured to perform routing functions.

Notes:

- sysctl settings are defined through files in `/usr/local/lib`, `/usr/lib/`, `/lib/`, `/run/`, and `/etc/`
- Files are typically placed in the `sysctl.d` directory within the parent directory
- The paths where sysctl preload files usually exist
  - `/run/sysctl.d/*.conf`
  - `/etc/sysctl.d/*.conf`
  - `/usr/local/lib/sysctl.d/*.conf`
  - `/usr/lib/sysctl.d/*.conf`
  - `/lib/sysctl.d/*.conf`
  - `/etc/sysctl.conf`
- Files must have the "`.conf`" extension
- Vendors settings usually live in `/usr/lib/` or `/usr/local/lib/`
- To override a whole file, create a new file with the same name in `/etc/sysctl.d/` and put new settings there.
- To override only specific settings, add a file with a lexically later name in `/etc/sysctl.d/` and put new settings there.
- The command `/usr/lib/systemd/systemd-sysctl --cat-config` produces output containing the system's loaded kernel parameters and the files they're configured in:
  - Entries listed later in the file take precedence over the same settings listed earlier in the file
  - Files containing kernel parameters that are over-ridden by other files with the same name will not be listed
  - On systems running UncomplicatedFirewall, the kernel parameters may be set or over-written. This will not be visible in the output of the command
- On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`
  - The settings in `/etc/ufw/sysctl.conf` will override settings other settings and **will not** be visible in the output of the `/usr/lib/systemd/systemd-sysctl --cat-config` command
  - This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

The system's loaded kernel parameters and the files they're configured in can be viewed by running the following command:

```
# /usr/lib/systemd/systemd-sysctl --cat-config
```

3.1.1 Ensure packet redirect sending is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

ICMP Redirects are used to send routing information to other hosts. As a host itself does not act as a router (in a host only configuration), there is no need to send redirects.

Rationale:

An attacker could use a compromised host to send invalid ICMP redirects to other router devices in an attempt to corrupt routing and have users access a system set up by the attacker as opposed to a valid system.

Impact:

IP forwarding is required on systems configured to act as a router. If these parameters are disabled, the system will not be able to perform as a router.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.conf.all.send_redirects` is set to `0`
- `net.ipv4.conf.default.send_redirects` is set to `0`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.conf.all.send_redirects=0"
"net.ipv4.conf.default.send_redirects=0")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
```

```
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.conf.all.send_redirects = 0`
- `net.ipv4.conf.default.send_redirects = 0`

Example:

```
# printf '%s\n' "net.ipv4.conf.all.send_redirects = 0"
"net.ipv4.conf.default.send_redirects = 0" >> /etc/sysctl.d/60-
netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.conf.all.send_redirects=0
   sysctl -w net.ipv4.conf.default.send_redirects=0
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.conf.all.send_redirects = 1

net.ipv4.conf.default.send_redirects = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.2 Ensure bogus icmp responses are ignored (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Setting `net.ipv4.icmp_ignore_bogus_error_responses` to `1` prevents the kernel from logging bogus responses (RFC-1122 non-compliant) from broadcast reframes, keeping file systems from filling up with useless log messages.

Rationale:

Some routers (and some attackers) will send responses that violate RFC-1122 and attempt to fill up a log file system with many useless error messages.

Audit:

Run the following script to verify the following kernel parameter is set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.icmp_ignore_bogus_error_responses` is set to `1`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.icmp_ignore_bogus_error_responses=1")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
```

```
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameter in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.icmp_ignore_bogus_error_responses = 1`

Example:

```
# printf '%s\n' "net.ipv4.icmp_ignore_bogus_error_responses = 1" >>
/etc/sysctl.d/60-netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.icmp_ignore_bogus_error_responses=1
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.icmp_ignore_bogus_error_responses = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.3 Ensure broadcast icmp requests are ignored (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Setting `net.ipv4.icmp_echo_ignore_broadcasts` to `1` will cause the system to ignore all ICMP echo and timestamp requests to broadcast and multicast addresses.

Rationale:

Accepting ICMP echo and timestamp requests with broadcast or multicast destinations for your network could be used to trick your host into starting (or participating) in a Smurf attack. A Smurf attack relies on an attacker sending large amounts of ICMP broadcast messages with a spoofed source address. All hosts receiving this message and responding would send echo-reply messages back to the spoofed address, which is probably not routable. If many hosts respond to the packets, the amount of traffic on the network could be significantly multiplied.

Audit:

Run the following script to verify the following kernel parameter is set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.icmp_echo_ignore_broadcasts` is set to `1`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.icmp_echo_ignore_broadcasts=1")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
```

```
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameter in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.icmp_echo_ignore_broadcasts = 1`

Example:

```
# printf '%s\n' "net.ipv4.icmp_echo_ignore_broadcasts = 1" >>
/etc/sysctl.d/60-netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.icmp_echo_ignore_broadcasts=1
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.icmp_echo_ignore_broadcasts = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.4 Ensure icmp redirects are not accepted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

ICMP redirect messages are packets that convey routing information and tell your host (acting as a router) to send packets via an alternate path. It is a way of allowing an outside routing device to update your system routing tables.

Rationale:

ICMP redirect messages are packets that convey routing information and tell your host (acting as a router) to send packets via an alternate path. It is a way of allowing an outside routing device to update your system routing tables. By setting `net.ipv4.conf.all.accept_redirects`, `net.ipv4.conf.default.accept_redirects`, `net.ipv6.conf.all.accept_redirects`, and `net.ipv6.conf.default.accept_redirects` to `0`, the system will not accept any ICMP redirect messages, and therefore, won't allow outsiders to update the system's routing tables.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.conf.all.accept_redirects` is set to `0`
- `net.ipv4.conf.default.accept_redirects` is set to `0`
- `net.ipv6.conf.all.accept_redirects` is set to `0`
- `net.ipv6.conf.default.accept_redirects` is set to `0`

Note:

- kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.
- IPv6 kernel parameters only apply to systems where IPv6 is enabled.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.conf.all.accept_redirects=0"
"net.ipv4.conf.default.accept_redirects=0"
"net.ipv6.conf.all.accept_redirects=0"
"net.ipv6.conf.default.accept_redirects=0")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
```

```
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.conf.all.accept_redirects = 0`
- `net.ipv4.conf.default.accept_redirects = 0`

Example:

```
# printf '%s\n' "net.ipv4.conf.all.accept_redirects = 0"
"net.ipv4.conf.default.accept_redirects = 0" >> /etc/sysctl.d/60-
netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.conf.all.accept_redirects=0
   sysctl -w net.ipv4.conf.default.accept_redirects=0
   sysctl -w net.ipv4.route.flush=1
}
```

- IF - IPv6 is enabled on the system:
Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv6.conf.all.accept_redirects = 0`
- `net.ipv6.conf.default.accept_redirects = 0`

Example:

```
# printf '%s\n' "net.ipv6.conf.all.accept_redirects = 0"
"net.ipv6.conf.default.accept_redirects = 0" >> /etc/sysctl.d/60-
netipv6_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv6.conf.all.accept_redirects=0
   sysctl -w net.ipv6.conf.default.accept_redirects=0
   sysctl -w net.ipv6.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.conf.all.accept_redirects = 1

net.ipv4.conf.default.accept_redirects = 1

net.ipv6.conf.all.accept_redirects = 1

net.ipv6.conf.default.accept_redirects = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.5 Ensure secure icmp redirects are not accepted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Secure ICMP redirects are the same as ICMP redirects, except they come from gateways listed on the default gateway list. It is assumed that these gateways are known to your system, and that they are likely to be secure.

Rationale:

It is still possible for even known gateways to be compromised. Setting `net.ipv4.conf.all.secure_redirects` and `net.ipv4.conf.default.secure_redirects` to `0` protects the system from routing table updates by possibly compromised known gateways.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.conf.all.secure_redirects` is set to `0`
- `net.ipv4.conf.default.secure_redirects` is set to `0`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.conf.all.secure_redirects=0"
"net.ipv4.conf.default.secure_redirects=0")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
```

```
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.conf.all.secure_redirects = 0`
- `net.ipv4.conf.default.secure_redirects = 0`

Example:

```
# printf '%s\n' "net.ipv4.conf.all.secure_redirects = 0"
"net.ipv4.conf.default.secure_redirects = 0" >> /etc/sysctl.d/60-
netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.conf.all.secure_redirects=0
   sysctl -w net.ipv4.conf.default.secure_redirects=0
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.conf.all.secure_redirects = 1

net.ipv4.conf.default.secure_redirects = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.6 Ensure reverse path filtering is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Setting `net.ipv4.conf.all.rp_filter` and `net.ipv4.conf.default.rp_filter` to `1` forces the Linux kernel to utilize reverse path filtering on a received packet to determine if the packet was valid. Essentially, with reverse path filtering, if the return packet does not go out the same interface that the corresponding source packet came from, the packet is dropped (and logged if `log_martians` is set).

Rationale:

Setting `net.ipv4.conf.all.rp_filter` and `net.ipv4.conf.default.rp_filter` to `1` is a good way to deter attackers from sending your system bogus packets that cannot be responded to. One instance where this feature breaks down is if asymmetrical routing is employed. This would occur when using dynamic routing protocols (bgp, ospf, etc) on your system. If you are using asymmetrical routing on your system, you will not be able to enable this feature without breaking the routing.

Impact:

If you are using asymmetrical routing on your system, you will not be able to enable this feature without breaking the routing.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.conf.all.rp_filter` is set to `1`
- `net.ipv4.conf.default.rp_filter` is set to `1`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.conf.all.rp_filter=1"
"net.ipv4.conf.default.rp_filter=1")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
```

```
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.conf.all.rp_filter = 1`
- `net.ipv4.conf.default.rp_filter = 1`

Example:

```
# printf '%s\n' "net.ipv4.conf.all.rp_filter = 1"
"net.ipv4.conf.default.rp_filter = 1" >> /etc/sysctl.d/60-netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.conf.all.rp_filter=1
   sysctl -w net.ipv4.conf.default.rp_filter=1
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.conf.all.rp_filter = 2

net.ipv4.conf.default.rp_filter = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.7 Ensure source routed packets are not accepted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

In networking, source routing allows a sender to partially or fully specify the route packets take through a network. In contrast, non-source routed packets travel a path determined by routers in the network. In some cases, systems may not be routable or reachable from some locations (e.g. private addresses vs. Internet routable), and so source routed packets would need to be used.

Rationale:

Setting `net.ipv4.conf.all.accept_source_route`, `net.ipv4.conf.default.accept_source_route`, `net.ipv6.conf.all.accept_source_route` and `net.ipv6.conf.default.accept_source_route` to `0` disables the system from accepting source routed packets. Assume this system was capable of routing packets to Internet routable addresses on one interface and private addresses on another interface. Assume that the private addresses were not routable to the Internet routable addresses and vice versa. Under normal routing circumstances, an attacker from the Internet routable addresses could not use the system as a way to reach the private address systems. If, however, source routed packets were allowed, they could be used to gain access to the private address systems as the route could be specified, rather than rely on routing protocols that did not allow this routing.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.conf.all.accept_source_route` is set to `0`
- `net.ipv4.conf.default.accept_source_route` is set to `0`
- `net.ipv6.conf.all.accept_source_route` is set to `0`
- `net.ipv6.conf.default.accept_source_route` is set to `0`

Note:

- kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.
- IPv6 kernel parameters only apply to systems where IPv6 is enabled.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.conf.all.accept_source_route=0"
"net.ipv4.conf.default.accept_source_route=0"
"net.ipv6.conf.all.accept_source_route=0"
"net.ipv6.conf.default.accept_source_route=0")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
```

```
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.conf.all.accept_source_route = 0`
- `net.ipv4.conf.default.accept_source_route = 0`

Example:

```
# printf '%s\n' "net.ipv4.conf.all.accept_source_route = 0"
"net.ipv4.conf.default.accept_source_route = 0" >> /etc/sysctl.d/60-
netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.conf.all.accept_source_route=0
   sysctl -w net.ipv4.conf.default.accept_source_route=0
   sysctl -w net.ipv4.route.flush=1
}
```

- IF - IPv6 is enabled on the system:
Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv6.conf.all.accept_source_route = 0`
- `net.ipv6.conf.default.accept_source_route = 0`

Example:

```
# printf '%s\n' "net.ipv6.conf.all.accept_source_route = 0"
"net.ipv6.conf.default.accept_source_route = 0" >> /etc/sysctl.d/60-
netipv6_sysctl.conf
```

Run the following command to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv6.conf.all.accept_source_route=0
   sysctl -w net.ipv6.conf.default.accept_source_route=0
   sysctl -w net.ipv6.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.conf.all.accept_source_route = 0

net.ipv4.conf.default.accept_source_route = 0

net.ipv6.conf.all.accept_source_route = 0

net.ipv6.conf.default.accept_source_route = 0

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.8 Ensure suspicious packets are logged (Automated)

Profile Applicability:

- Level 1 - Server

Description:

When enabled, this feature logs packets with un-routable source addresses to the kernel log.

Rationale:

Setting `net.ipv4.conf.all.log_martians` and `net.ipv4.conf.default.log_martians` to `1` enables this feature. Logging these packets allows an administrator to investigate the possibility that an attacker is sending spoofed packets to their system.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.conf.all.log_martians` is set to `1`
- `net.ipv4.conf.default.log_martians` is set to `1`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.conf.all.log_martians=1"
"net.ipv4.conf.default.log_martians=1")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
```

```
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.conf.all.log_martians = 1`
- `net.ipv4.conf.default.log_martians = 1`

Example:

```
# printf '%s\n' "net.ipv4.conf.all.log_martians = 1"
"net.ipv4.conf.default.log_martians = 1" >> /etc/sysctl.d/60-
netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.conf.all.log_martians=1
   sysctl -w net.ipv4.conf.default.log_martians=1
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.conf.all.log_martians = 0

net.ipv4.conf.default.log_martians = 0

References:

1. NIST SP 800-53 Rev. 5: AU-3

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.5 Collect Detailed Audit Logs Configure detailed audit logging for enterprise assets containing sensitive data. Include event source, date, username, timestamp, source addresses, destination addresses, and other useful elements that could assist in a forensic investigation.		●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

3.1.9 Ensure tcp syn cookies is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

When `tcp_syncookies` is set, the kernel will handle TCP SYN packets normally until the half-open connection queue is full, at which time, the SYN cookie functionality kicks in. SYN cookies work by not using the SYN queue at all. Instead, the kernel simply replies to the SYN with a SYN/ACK, but will include a specially crafted TCP sequence number that encodes the source and destination IP address and port number and the time the packet was sent. A legitimate connection would send the ACK packet of the three way handshake with the specially crafted sequence number. This allows the system to verify that it has received a valid response to a SYN cookie and allow the connection, even though there is no corresponding SYN in the queue.

Rationale:

Attackers use SYN flood attacks to perform a denial of service attacked on a system by sending many SYN packets without completing the three way handshake. This will quickly use up slots in the kernel's half-open connection queue and prevent legitimate connections from succeeding. Setting `net.ipv4.tcp_syncookies` to `1` enables SYN cookies, allowing the system to keep accepting valid connections, even if under a denial of service attack.

Audit:

Run the following script to verify the following kernel parameter is set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv4.tcp_syncookies` is set to `1`

Note: kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv4.tcp_syncookies=1")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
```

```
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Set the following parameter in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv4.tcp_syncookies = 1`

Example:

```
# printf '%s\n' "net.ipv4.tcp_syncookies = 1" >> /etc/sysctl.d/60-
netipv4_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv4.tcp_syncookies=1
   sysctl -w net.ipv4.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv4.tcp_syncookies = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

3.1.10 Ensure ipv6 router advertisements are not accepted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Routers periodically multicast Router Advertisement messages to announce their availability and convey information to neighboring nodes that enable them to be automatically configured on the network.

`net.ipv6.conf.all.accept_ra` and `net.ipv6.conf.default.accept_ra` determine the systems ability to accept these advertisements.

Rationale:

It is recommended that systems do not accept router advertisements as they could be tricked into routing traffic to compromised machines. Setting hard routes within the system (usually a single default route to a trusted router) protects the system from bad routes. Setting `net.ipv6.conf.all.accept_ra` and `net.ipv6.conf.default.accept_ra` to `0` disables the system's ability to accept IPv6 router advertisements.

Audit:

Run the following script to verify the following kernel parameters are set in the running configuration and correctly loaded from a kernel parameter configuration file:

- `net.ipv6.conf.all.accept_ra` is set to `0`
- `net.ipv6.conf.default.accept_ra` is set to `0`

Note:

- kernel parameters are loaded by file and parameter order precedence. The following script observes this precedence as part of the auditing procedure. The parameters being checked may be set correctly in a file. If that file is superseded, the parameter is overridden by an incorrect setting later in that file, or in a canonically later file, that "correct" setting will be ignored both by the script and by the system during a normal kernel parameter load sequence.
- IPv6 kernel parameters only apply to systems where IPv6 is enabled.

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ipv6_disabled=""
   a_parlist=("net.ipv6.conf.all.accept_ra=0"
"net.ipv6.conf.default.accept_ra=0")
   l_ufwscf="$([ -f /etc/default/ufw ] && awk -F= '/^\s*IPT_SYSCTL=/ {print
$2}' /etc/default/ufw)"
   f_ipv6_chk()
   {
      l_ipv6_disabled="no"
      ! grep -Pqs -- '^\h*0\b' /sys/module/ipv6/parameters/disable &&
l_ipv6_disabled="yes"
      if sysctl net.ipv6.conf.all.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.all\.disable_ipv6\h*=\h*1\b" && \
         sysctl net.ipv6.conf.default.disable_ipv6 | grep -Pqs --
"^\h*net\.ipv6\.conf\.default\.disable_ipv6\h*=\h*1\b"; then
         l_ipv6_disabled="yes"
      fi
   }
   f_kernel_parameter_chk()
   {
      l_running_parameter_value="$(sysctl "$l_parameter_name" | awk -F=
'{print $2}' | xargs)" # Check running configuration
      if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_running_parameter_value"; then
         a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_running_parameter_value\""
         "    in the running configuration")
      else
         a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_running_parameter_value\"" \
         "    in the running configuration" \
         "    and should have a value of: \"$l_value_out\"")
      fi
      unset A_out; declare -A A_out # Check durable setting (files)
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_kpar="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_file")
            fi
         fi
      done < <("$l_systemdsysctl" --cat-config | grep -Po
'^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if [ -n "$l_ufwscf" ]; then # Account for systems with UFW (Not covered
by systemd-sysctl --cat-config)
         l_kpar="$(grep -Po "^\h*$l_parameter_name\b" "$l_ufwscf" | xargs)"
         l_kpar="${l_kpar//\//.}"
         [ "$l_kpar" = "$l_parameter_name" ] &&
A_out+=(["$l_kpar"]="$l_ufwscf")
      fi
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate
output
```

```
         while IFS="=" read -r l_fkpname l_file_parameter_value; do
            l_fkpname="${l_fkpname// /}";
l_file_parameter_value="${l_file_parameter_value// /}"
            if grep -Pq -- '\b'"$l_parameter_value"'\b' <<<
"$l_file_parameter_value"; then
               a_output+=(" - \"$l_parameter_name\" is correctly set to
\"$l_file_parameter_value\"" \
               "    in \"$(printf '%s' "${A_out[@]}")\"")
            else
               a_output2+=(" - \"$l_parameter_name\" is incorrectly set to
\"$l_file_parameter_value\""
               "    in \"$(printf '%s' "${A_out[@]}")\"" \
               "    and should have a value of: \"$l_value_out\"")
            fi
         done < <(grep -Po -- "^\h*$l_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         a_output2+=(" - \"$l_parameter_name\" is not set in an included
file" \
         "    ** Note: \"$l_parameter_name\" May be set in a file that's
ignored by load procedure **")
      fi
   }
   l_systemdsysctl="$(readlink -f /lib/systemd/systemd-sysctl)"
   while IFS="=" read -r l_parameter_name l_parameter_value; do # Assess and
check parameters
      l_parameter_name="${l_parameter_name// /}";
l_parameter_value="${l_parameter_value// /}"
      l_value_out="${l_parameter_value//-/ through }";
l_value_out="${l_value_out//|/ or }"
      l_value_out="$(tr -d '(){}' <<< "$l_value_out")"
      if grep -q '^net.ipv6.' <<< "$l_parameter_name"; then
         [ -z "$l_ipv6_disabled" ] && f_ipv6_chk
         if [ "$l_ipv6_disabled" = "yes" ]; then
            a_output+=(" - IPv6 is disabled on the system,
\"$l_parameter_name\" is not applicable")
         else
            f_kernel_parameter_chk
         fi
      else
         f_kernel_parameter_chk
      fi
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

- IF - IPv6 is enabled on the system:
Set the following parameters in `/etc/sysctl.conf` or a file in `/etc/sysctl.d/` ending in `.conf`:

- `net.ipv6.conf.all.accept_ra = 0`
- `net.ipv6.conf.default.accept_ra = 0`

Example:

```
# printf '%s\n' "net.ipv6.conf.all.accept_ra = 0"
"net.ipv6.conf.default.accept_ra = 0" >> /etc/sysctl.d/60-netipv6_sysctl.conf
```

Run the following script to set the active kernel parameters:

```
#!/usr/bin/env bash

{
   sysctl -w net.ipv6.conf.all.accept_ra=0
   sysctl -w net.ipv6.conf.default.accept_ra=0
   sysctl -w net.ipv6.route.flush=1
}
```

Note: If these settings appear in a canonically later file, or later in the same file, these settings will be overwritten.

Default Value:

net.ipv6.conf.all.accept_ra = 1

net.ipv6.conf.default.accept_ra = 1

References:

1. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5

Additional Information:

On systems with Uncomplicated Firewall, additional settings may be configured in `/etc/ufw/sysctl.conf`

- The settings in `/etc/ufw/sysctl.conf` will override settings in `/etc/sysctl.conf`
- This behavior can be changed by updating the `IPT_SYSCTL` parameter in `/etc/default/ufw`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

4 Host Based Firewall

A Host Based Firewall, on a Linux system, is a set of rules used to protect machines from any unwanted traffic from outside. It enables users to control incoming network traffic on host machines by defining a set of firewall rules. These rules are used to sort the incoming traffic and either block it or allow it through.

To provide a Host Based Firewall, the Linux kernel includes support for:

- Netfilter (IPTables) - A set of hooks inside the Linux kernel that allows kernel modules to register callback functions with the network stack. A registered callback function is then called back for every packet that traverses the respective hook within the network stack. Includes the ip_tables, ip6_tables, arp_tables, and ebtables kernel modules. These modules are some of the significant parts of the Netfilter hook system.
- NFTables - A subsystem of the Linux kernel providing filtering and classification of network packets/datagrams/frames. nftables is supposed to replace certain parts of Netfilter, while keeping and reusing most of it. nftables utilizes the building blocks of the Netfilter infrastructure, such as the existing hooks into the networking stack, connection tracking system, userspace queueing component, and logging subsystem. Is available in Linux kernels 3.13 and newer.

In order to configure firewall rules for Netfilter or nftables, a firewall utility needs to be installed. Though Linux does have multiple firewall configurations utilities listed bellow available, in instances like AKS, the intent of the system is to act as a container host. Because of this, the firewall is expected to be configured through the container software.

- IPTables - Includes the iptables, ip6tables, arptables and ebtables utilities for configuration Netfilter and the ip_tables, ip6_tables, arp_tables, and ebtables kernel modules.
- NFTables - Includes the nft utility for configuration of the nftables subsystem of the Linux kernel
- FirewallD - Provides firewall features by acting as a front-end for the Linux kernel's netfilter framework via the iptables backend. Starting in v0.6.0, FirewallD added support for acting as a front-end for the Linux kernel's netfilter framework via the nftables userspace utility, acting as an alternative to the nft command line program. firewalld supports both IPv4 and IPv6 networks and can administer separate firewall zones with varying degrees of trust as defined in zone profiles.

We highly recommend the following be included in the final configuration of the firewall:

- Inbound is configured as deny all.
- Forward is configured as deny all.
- Outbound is configured as deny all.

Note: Only one method should be used to configure a firewall on the system. Use of more than one method could produce unexpected results.

4.1 Configure host based firewall packages

A Host based firewall package is required to configure a firewall of the system

4.1.1 Ensure iptables is installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

iptables is a utility program that allows a system administrator to configure the tables provided by the Linux kernel firewall, implemented as different Netfilter modules, and the chains and rules it stores. Different kernel modules and programs are used for different protocols; iptables applies to IPv4, ip6tables to IPv6, arptables to ARP, and ebtables to Ethernet frames.

Rationale:

A method of configuring and maintaining firewall rules is necessary to configure a Host Based Firewall.

Audit:

Run the following command to verify that `iptables` is installed:

```
# rpm -q iptables

iptables-<version>
```

Remediation:

Run the following command to install `iptables`:

```
# tdnf install iptables
```

References:

1. NIST SP 800-53 Rev. 5: CM-7, CA-9

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.4 Implement and Manage a Firewall on Servers Implement and manage a firewall on servers, where supported. Example implementations include a virtual firewall, operating system firewall, or a third-party firewall agent.	●	●	●
v7	9.4 Apply Host-based Firewalls or Port Filtering Apply host-based firewalls or port filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	●	●	●

4.1.2 Ensure nftables is not in use (Automated)

Profile Applicability:

- Level 1 - Server

Description:

nftables provides a new in-kernel packet classification framework that is based on a network-specific Virtual Machine (VM) and a new nft userspace command line tool.

nftables reuses the existing Netfilter subsystems such as the existing hook infrastructure, the connection tracking system, NAT, userspace queuing and logging subsystem.

Rationale:

AKS nodes leverage Netfilter (IPTables) to provide a host based firewall.

Running multiple firewalls on a system can produce unexpected results.

Impact:

Changing firewall settings while connected over the network can result in being locked out of the system.

Audit:

Run the following command to determine if the `nftables` package is installed:

```
# rpm -q nftables &>/dev/null && echo "nftables is installed"
```

Verify that the NFTables package is not installed.
- OR -
If the NFTables package is required for dependencies, run the following commands to verify that nftables.service is not enabled and not active:
Run the following command to verify nftables.service is not enabled:

```
# systemctl is-enabled nftables.service
```

Verify the output is not enabled.
Run the following command to verify nftables.service is not active:

```
# systemctl is-active nftables.service

inactive
```

Remediation:

Ensure either IPTables is being used, `nftables.service` is not enabled and not active
- OR -
If nftables is being used; it is installed, enabled, and active.
Run the following command to remove the NFTables package:

```
# tdnf remove nftables
```

- OR- If the NFTables package is required for a dependency:
Run the following command to mask nftables.service:

```
# systemctl mask nftables.service
```

Run the following command to stop nftables.service:

```
# systemctl stop nftables.service
```

References:

1. NIST SP 800-53 Rev. 5: CA-9

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.4 Implement and Manage a Firewall on Servers Implement and manage a firewall on servers, where supported. Example implementations include a virtual firewall, operating system firewall, or a third-party firewall agent.	●	●	●
v7	9.4 Apply Host-based Firewalls or Port Filtering Apply host-based firewalls or port filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	●	●	●

4.1.3 Ensure firewalld is not in use (Automated)

Profile Applicability:

- Level 1 - Server

Description:

In Linux security, employing a single, effective firewall configuration utility is crucial. Firewalls act as digital gatekeepers by filtering network traffic based on rules. Proper firewall configurations ensure that only legitimate traffic gets processed, reducing the system's exposure to potential threats. The choice between FirewallD and NFTables depends on organizational specific needs:

`FirewallD` - Is a firewall service daemon that provides a dynamic customizable host-based firewall with a D-Bus interface. Being dynamic, it enables creating, changing, and deleting the rules without the necessity to restart the firewall daemon each time the rules are changed.

`NFTables` - Includes the nft utility for configuration of the nftables subsystem of the Linux kernel.

Notes:

- firewalld with nftables backend does not support passing custom nftables rules to firewalld, using the `--direct` option.
- In order to configure firewall rules for nftables, a firewall utility needs to be installed and active of the system. The use of more than one firewall utility may produce unexpected results.
- Allow port 22(ssh) needs to be updated to only allow systems requiring ssh connectivity to connect, as per site policy.

Rationale:

Proper configuration of a single firewall utility minimizes cyber threats and protects services and data, while avoiding vulnerabilities like open ports or exposed services. Standardizing on a single tool simplifies management, reduces errors, and fortifies security across Linux systems.

AKS nodes do not currently fully support the use of FirewallD. If installed, FirewallD should be either removed or stopped and masked to prevent unexpected results.

Audit:

Run the following command to verify FirewallD is not installed:

```
# rpm -q firewalld

package firewalld is not installed
```

- OR - If the firewalld package is required for a dependency:
Run the following command to verify firewalld.service is not enabled:

```
# systemctl is-enabled firewalld.service
```

Verify the output is not enabled
Run the following command to verify firewalld.service is not active:

```
# systemctl is-active firewalld.service

inactive
```

Remediation:

Run the following command to remove the FirewallD package:

```
# tdnf remove firewalld
```

- OR - If the firewalld package is required for a dependency:
Run the following command to mask firewalld.service:

```
# systemctl mask firewalld.service
```

Run the following command to stop firewalld.service:

```
# systemctl stop firewalld
```

References:

1. https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html-single/configuring_firewalls_and_packet_filters/index

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.4 Implement and Manage a Firewall on Servers Implement and manage a firewall on servers, where supported. Example implementations include a virtual firewall, operating system firewall, or a third-party firewall agent.	●	●	●
v8	4.5 Implement and Manage a Firewall on End-User Devices Implement and manage a host-based firewall or port-filtering tool on end-user devices, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	●	●	●
v7	9.4 Apply Host-based Firewalls or Port Filtering Apply host-based firewalls or port filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	●	●	●

5 Access, Authentication and Authorization

5.1 Configure time-based job schedulers

`cron` is a time-based job scheduler used to schedule jobs, commands or shell scripts, to run periodically at fixed times, dates, or intervals.

`at` provides the ability to execute a command or shell script at a specified date and hour, or after a given interval of time.

Other methods exist for scheduling jobs, such as `systemd timers`. If another method is used, it should be secured in accordance with local site policy

Note: `systemd timers` are systemd unit files whose name ends in `.timer` that control `.service` files or events. Timers can be used as an alternative to `cron` and `at`. Timers have built-in support for calendar time events, monotonic time events, and can be run asynchronously

If `cron` and `at` are not installed, this section can be skipped.

5.1.1 Ensure cron daemon is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `cron` daemon is used to execute batch jobs on the system.

Rationale:

While there may not be user jobs that need to be run on the system, the system does have maintenance jobs that may include security monitoring that have to run, and `cron` is used to execute them.

Audit:

Run the the following command to verify `cron` is enabled:

```
# systemctl is-enabled crond

enabled
```

Verify result is "enabled".

Remediation:

Run the following command to enable `cron`:

```
# systemctl --now enable crond
```

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

Additional methods of enabling a service exist. Consult your distribution documentation for appropriate methods.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

5.1.2 Ensure permissions on /etc/crontab are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/crontab` file is used by `cron` to control its own jobs. The commands in this item make sure that root is the user and group owner of the file and that only the owner can access the file.

Rationale:

This file contains information on what system jobs are run by cron. Write access to these files could provide unprivileged users with the ability to elevate their privileges. Read access to these files could provide users with the ability to gain insight on system jobs that run on the system and could provide them a way to gain unauthorized privileged access.

Audit:

Run the following command and verify `Uid` and `Gid` are both `0/root` and `Access` does not grant permissions to `group` or `other` :

```
# stat /etc/crontab
Access: (0600/-rw-------)  Uid: (    0/    root)   Gid: (    0/    root)
```

Remediation:

Run the following commands to set ownership and permissions on `/etc/crontab` :

```
# chown root:root /etc/crontab
# chmod og-rwx /etc/crontab
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.3 Ensure permissions on /etc/cron.hourly are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

This directory contains system `cron` jobs that need to run on an hourly basis. The files in this directory cannot be manipulated by the `crontab` command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Run the following command and verify `Uid` and `Gid` are both `0/root` and `Access` does not grant permissions to `group` or `other` :

```
# stat /etc/cron.hourly

Access: (0700/drwx------)  Uid: (    0/    root)   Gid: (    0/    root)
```

Remediation:

Run the following commands to set ownership and permissions on `/etc/cron.hourly` :

```
# chown root:root /etc/cron.hourly

# chmod og-rwx /etc/cron.hourly
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.4 Ensure permissions on /etc/cron.daily are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/cron.daily` directory contains system cron jobs that need to run on a daily basis. The files in this directory cannot be manipulated by the `crontab` command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Run the following command and verify `Uid` and `Gid` are both `0/root` and `Access` does not grant permissions to `group` or `other` :

```
# stat /etc/cron.daily

Access: (0700/drwx------)  Uid: (    0/    root)   Gid: (    0/    root)
```

Remediation:

Run the following commands to set ownership and permissions on `/etc/cron.daily` :

```
# chown root:root /etc/cron.daily

# chmod og-rwx /etc/cron.daily
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.5 Ensure permissions on /etc/cron.weekly are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/cron.weekly` directory contains system cron jobs that need to run on a weekly basis. The files in this directory cannot be manipulated by the `crontab` command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Run the following command and verify `Uid` and `Gid` are both `0/root` and `Access` does not grant permissions to `group` or `other` :

```
# stat /etc/cron.weekly

Access: (0700/drwx------)  Uid: (    0/    root)   Gid: (    0/    root)
```

Remediation:

Run the following commands to set ownership and permissions on `/etc/cron.weekly` :

```
# chown root:root /etc/cron.weekly

# chmod og-rwx /etc/cron.weekly
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.6 Ensure permissions on /etc/cron.monthly are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/cron.monthly` directory contains system cron jobs that need to run on a monthly basis. The files in this directory cannot be manipulated by the `crontab` command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Run the following command and verify `Uid` and `Gid` are both `0/root` and `Access` does not grant permissions to `group` or `other` :

```
# stat /etc/cron.monthly

Access: (0700/drwx------)  Uid: (    0/    root)   Gid: (    0/    root)
```

Remediation:

Run the following commands to set ownership and permissions on `/etc/cron.monthly` :

```
# chown root:root /etc/cron.monthly

# chmod og-rwx /etc/cron.monthly
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.7 Ensure permissions on /etc/cron.d are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/cron.d` directory contains system `cron` jobs that need to run in a similar manner to the hourly, daily weekly and monthly jobs from `/etc/crontab` , but require more granular control as to when they run. The files in this directory cannot be manipulated by the `crontab` command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Run the following command and verify `Uid` and `Gid` are both `0/root` and `Access` does not grant permissions to `group` or `other` :

```
# stat /etc/cron.d

Access: (0700/drwx------)  Uid: (    0/    root)   Gid: (    0/    root)
```

Remediation:

Run the following commands to set ownership and permissions on `/etc/cron.d` :

```
# chown root:root /etc/cron.d

# chmod og-rwx /etc/cron.d
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.8 Ensure cron is restricted to authorized users (Automated)

Profile Applicability:

- Level 1 - Server

Description:

If `cron` is installed in the system, configure `/etc/cron.allow` to allow specific users to use these services. If `/etc/cron.allow` does not exist, then `/etc/cron.deny` is checked. Any user not specifically defined in those files is allowed to use cron. If both `/etc/cron.allow` and `/etc/cron.deny` exist, or only `/etc/cron.allow` exists, only users in `/etc/cron.allow` are allowed to use cron.

Note: Even though a given user is not listed in `cron.allow`, cron jobs can still be run as that user. The `cron.allow` file only controls administrative access to the crontab command for scheduling and modifying cron jobs.

Rationale:

On many systems, only the system administrator is authorized to schedule `cron` jobs. Using the `cron.allow` file to control who can run `cron` jobs enforces this policy. It is easier to manage an allow list than a deny list. In a deny list, you could potentially add a user ID to the system and forget to add it to the deny files.

Audit:

Run the following command to verify `/etc/cron.allow` exists, is mode `0640` or more restrictive, is owned by `root`, and group owned by `root`:

```
# stat -Lc 'File: (%n) Access: (%#a/%A) Uid: ( %u/ %U) Gid: ( %g/ %G)'
/etc/cron.allow

File: (/etc/cron.allow) Access: (0640/-rw-r-----) Uid: ( 0/ root) Gid: ( 0/
root)
```

Run the following command to verify `/etc/cron.deny` doesn't exist, or: is mode `0640` or more restrictive, is owned by `root`, and group owned by `root`:

```
# stat -Lc 'File: (%n) Access: (%#a/%A) Uid: ( %u/ %U) Gid: ( %g/ %G)'
/etc/cron.deny

stat: cannot stat '/etc/cron.deny': No such file or directory
 -OR-
File: (/etc/cron.deny) Access: (0640/-rw-r-----) Uid: ( 0/ root) Gid: ( 0/
root)
```

Remediation:

Run the following script to remove `/etc/cron.deny`, create `/etc/cron.allow`, and set the file mode on `/etc/cron.allow`:

```
#!/usr/bin/env bash

{
   if rpm -q cronie >/dev/null; then
      [ -e /etc/cron.deny ] && rm -f /etc/cron.deny
      [ ! -e /etc/cron.allow ] && touch /etc/cron.allow
      chown root:root /etc/cron.allow
      chmod g-wx,o-rwx /etc/cron.allow
   else
      echo "cron is not installed on the system"
   fi
}
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.1.9 Ensure at is restricted to authorized users (Automated)

Profile Applicability:

- Level 1 - Server

Description:

If `at` is installed in the system, configure `/etc/at.allow` to allow specific users to use these services. If `/etc/at.allow` does not exist, then `/etc/at.deny` is checked. Any user not specifically defined in those files is allowed to use `at`. By removing the file, only users in `/etc/at.allow` are allowed to use `at`.

Note: Even though a given user is not listed in `at.allow`, `at` jobs can still be run as that user. The `at.allow` file only controls administrative access to the `at` command for scheduling and modifying `at` jobs.

Rationale:

On many systems, only the system administrator is authorized to schedule `at` jobs. Using the `at.allow` file to control who can run `at` jobs enforces this policy. It is easier to manage an allow list than a deny list. In a deny list, you could potentially add a user ID to the system and forget to add it to the deny files.

Audit:

Run the following script:

```
#!/usr/bin/env bash

{
   if rpm -q at >/dev/null; then
      [ -e /etc/at.deny ] && echo "Fail: at.deny exists"
      if [ ! -e /etc/at.allow ]; then
         echo "Fail: at.allow doesn't exist"
      else
         ! stat -Lc "%a" /etc/at.allow | grep -Eq "[0,2,4,6]00" && echo
"Fail: at.allow mode too permissive"
         ! stat -Lc "%u:%g" /etc/at.allow | grep -Eq "^0:0$" && echo "Fail:
at.allow owner and/or group not root"
      fi
      if [ ! -e /etc/at.deny ] && [ -e /etc/at.allow ] && stat -Lc "%a"
/etc/at.allow | grep -Eq "[0,2,4,6]00" \
         && stat -Lc "%u:%g" /etc/at.allow | grep -Eq "^0:0$"; then
         echo "Pass"
      fi
   else
      echo "Pass: at is not installed on the system"
   fi
}
```

Verify the output of the script includes `Pass`.

Remediation:

Run the following script to remove `/etc/at.deny`, create `/etc/at.allow`, and set the file mode for `/etc/at.allow`:

```
#!/usr/bin/env bash

{
   if rpm -q at >/dev/null; then
      [ -e /etc/at.deny ] && rm -f /etc/at.deny
      [ ! -e /etc/at.allow ] && touch /etc/at.allow
      chown root:root /etc/at.allow
      chmod u-x,go-rwx /etc/at.allow
   else
      echo "at is not installed on the system"
   fi
}
```

OR Run the following command to remove `at`:

```
# tdnf remove at
```

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.2 Configure SSH Server

Secure Shell (SSH) is a secure, encrypted replacement for common login services such as `telnet`, `ftp`, `rlogin`, `rsh`, and `rcp`. It is strongly recommended that sites abandon older clear-text login protocols and use SSH to prevent session hijacking and sniffing of sensitive data off the network.

The recommendations in this section only apply if the SSH daemon is installed on the system, **if remote access is not required the SSH daemon can be removed and this section skipped**.

`sshd_config`:

- The openSSH daemon configuration directives, `Include` and `Match`, may cause the audits in this section's recommendations to report incorrectly. It is recommended that these options only be used if they're needed and fully understood. If these options are configured in accordance with local site policy, they should be accounted for when following the recommendations in this section.
- The default `Include` location is the `/etc/ssh/sshd_config.d` directory. This default has been accounted for in this section. If a file has an additional `Include` that isn't this default location, the files should be reviewed to verify that the recommended setting is not being over-ridden.
- The audits of the running configuration in this section are run in the context of the root user, the local host name, and the local host's IP address. If a `Match` block exists that matches one of these criteria, the output of the audit will be from the match block. The respective matched criteria should be replaced with a non-matching substitution.
- `Include`:
    - Include the specified configuration file(s).
    - Multiple pathnames may be specified and each pathname may contain glob(7) wildcards that will be expanded and processed in lexical order.
    - Files without absolute paths are assumed to be in `/etc/ssh/`.
    - An Include directive may appear inside a Match block to perform conditional inclusion.

- `Match`:
  - Introduces a conditional block. If all of the criteria on the Match line are satisfied, the keywords on the following lines override those set in the global section of the config file, until either another Match line or the end of the file. If a keyword appears in multiple Match blocks that are satisfied, only the first instance of the keyword is applied.
  - The arguments to Match are one or more criteria-pattern pairs or the single token All which matches all criteria. The available criteria are User, Group, Host, LocalAddress, LocalPort, and Address.
  - The match patterns may consist of single entries or comma-separated lists and may use the wildcard and negation operators described in the PATTERNS section of ssh_config(5).
  - The patterns in an Address criteria may additionally contain addresses to match in CIDR address/masklen format, such as `192.0.2.0/24` or `2001:db8::/32`. Note that the mask length provided must be consistent with the address - it is an error to specify a mask length that is too long for the address or one with bits set in this host portion of the address. For example, `192.0.2.0/33` and `192.0.2.0/8`, respectively.
  - Only a subset of keywords may be used on the lines following a Match keyword. Available keywords are available in the ssh_config man page.
- Once all configuration changes have been made to `/etc/ssh/sshd_config` or any included configuration files, the `sshd` configuration must be reloaded

Command to re-load the SSH daemon configuration:

```
# systemctl reload-or-restart sshd
```

`sshd` command:

- `-T` - Extended test mode. Check the validity of the configuration file, output the effective configuration to stdout and then exit. Optionally, Match rules may be applied by specifying the connection parameters using one or more `-C` options.
- `-C` - connection_spec. Specify the connection parameters to use for the -T extended test mode. If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr`, `user`, `host`, `laddr`, `lport`, and `rdomain` and correspond to source address, user, resolved source host name, local address, local port number and routing domain respectively.

5.2.1 Ensure access to /etc/ssh/sshd_config is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The file `/etc/ssh/sshd_config`, and files ending in `.conf` in the `/etc/ssh/sshd_config.d` directory, contain configuration specifications for `sshd`.

Rationale:

Configuration specifications for `sshd` need to be protected from unauthorized changes by non-privileged users.

Audit:

Run the following script and verify `/etc/ssh/sshd_config` and files ending in `.conf` in the `/etc/ssh/sshd_config.d` directory are:

- Mode `0600` or more restrictive
- Owned by the `root` user
- Group owned by the group `root`

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=()
   perm_mask='0177' && maxperm="$( printf '%o' $(( 0777 & ~$perm_mask)) )"
   f_sshd_files_chk()
   {
      while IFS=: read -r l_mode l_user l_group; do
         a_out2=()
         [ $(( $l_mode & $perm_mask )) -gt 0 ] && a_out2+=("    Is mode:
\"$l_mode\"" \
         "    should be mode: \"$maxperm\" or more restrictive")
         [ "$l_user" != "root" ] && a_out2+=("    Is owned by \"$l_user\"
should be owned by \"root\"")
         [ "$l_group" != "root" ] && a_out2+=("    Is group owned by
\"$l_user\" should be group owned by \"root\"")
         if [ "${#a_out2[@]}" -gt "0" ]; then
            a_output2+=("  - File: \"$l_file\":" "${a_out2[@]}")
         else
            a_output+=("  - File: \"$l_file\":" "    Correct: mode ($l_mode),
owner ($l_user)" \
            "    and group owner ($l_group) configured")
         fi
      done < <(stat -Lc '%#a:%U:%G' "$l_file")
   }
   [ -e "/etc/ssh/sshd_config" ] && l_file="/etc/ssh/sshd_config" &&
f_sshd_files_chk
   while IFS= read -r -d $'\0' l_file; do
      [ -e "$l_file" ] && f_sshd_files_chk
   done < <(find /etc/ssh/sshd_config.d -type f -name '*.conf' \( -perm /077
-o ! -user root -o ! -group root \) -print0 2>/dev/null)
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

- IF - other locations are listed in an `Include` statement, `*.conf` files in these locations should also be checked.

Remediation:

Run the following script to set ownership and permissions on `/etc/ssh/sshd_config` and files ending in `.conf` in the `/etc/ssh/sshd_config.d` directory:

```
#!/usr/bin/env bash

{
   chmod u-x,og-rwx /etc/ssh/sshd_config
   chown root:root /etc/ssh/sshd_config
   while IFS= read -r -d $'\0' l_file; do
      if [ -e "$l_file" ]; then
         chmod u-x,og-rwx "$l_file"
         chown root:root "$l_file"
      fi
   done < <(find /etc/ssh/sshd_config.d -type f -print0 2>/dev/null)
}
```

- IF - other locations are listed in an `Include` statement, `*.conf` files in these locations access should also be modified.

Default Value:

Access: (0600/-rw-------) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3. MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.2.2 Ensure access to SSH private host key files is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

An SSH private key is one of two files used in SSH public key authentication. In this authentication method, the possession of the private key is proof of identity. Only a private key that corresponds to a public key will be able to authenticate successfully. The private keys need to be stored and handled carefully, and no copies of the private key should be distributed.

Rationale:

If an unauthorized user obtains the private SSH host key file, the host could be impersonated.

Audit:

Run the following script to verify SSH private host key files are owned by the root user and either:

- owned by the group root and mode `0600` or more restrictive

- OR -

- owned by the group designated to own openSSH private keys and mode `0640` or more restrictive

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=()
   l_ssh_group_name="$(awk -F: '($1 ~ /^(ssh_keys|_?ssh)$/) {print $1}'
/etc/group)"
   f_file_chk()
   {
      while IFS=: read -r l_file_mode l_file_owner l_file_group; do
         a_out2=()
         [ "$l_file_group" = "$l_ssh_group_name" ] && l_pmask="0137" ||
l_pmask="0177"
         l_maxperm="$( printf '%o' $(( 0777 & ~$l_pmask )) )"
         if [ $(( $l_file_mode & $l_pmask )) -gt 0 ]; then
            a_out2+=("    Mode: \"$l_file_mode\" should be mode:
\"$l_maxperm\" or more restrictive")
         fi
         if [ "$l_file_owner" != "root" ]; then
            a_out2+=("    Owned by: \"$l_file_owner\" should be owned by
\"root\"")
         fi
         if [[ ! "$l_file_group" =~ ($l_ssh_group_name|root) ]]; then
            a_out2+=("    Owned by group \"$l_file_group\" should be group
owned by: \"$l_ssh_group_name\" or \"root\"")
         fi
         if [ "${#a_out2[@]}" -gt "0" ]; then
            a_output2+=("  - File: \"$l_file\"${a_out2[@]}")
         else
            a_output+=("  - File: \"$l_file\"" \
            "    Correct: mode: \"$l_file_mode\", owner: \"$l_file_owner\"
and group owner: \"$l_file_group\" configured")
         fi
      done < <(stat -Lc '%#a:%U:%G' "$l_file")
   }
   while IFS= read -r -d $'\0' l_file; do
      if ssh-keygen -lf &>/dev/null "$l_file"; then
         file "$l_file" | grep -Piq --
'\bopenssh\h+([^#\n\r]+\h+)?private\h+key\b' && f_file_chk
      fi
   done < <(find -L /etc/ssh -xdev -type f -print0 2>/dev/null)
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Run the following script to set mode, ownership, and group on the private SSH host key files:

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=(); l_ssh_group_name="$(awk -F: '($1 ~ /^(ssh_keys|_?ssh)$/)
{print $1}' /etc/group)"
   f_file_access_fix()
   {
      while IFS=: read -r l_file_mode l_file_owner l_file_group; do
         a_out2=()
         [ "$l_file_group" = "$l_ssh_group_name" ] && l_pmask="0137" || l_pmask="0177"
         l_maxperm="$( printf '%o' $(( 0777 & ~$l_pmask )) )"
         if [ $(( $l_file_mode & $l_pmask )) -gt 0 ]; then
            a_out2+=("    Mode: \"$l_file_mode\" should be mode: \"$l_maxperm\" or
more restrictive" \
            "    updating to mode: \:$l_maxperm\"")
            if [ "l_file_group" = "$l_ssh_group_name" ]; then
               chmod u-x,g-wx,o-rwx "$l_file"
            else
               chmod u-x,go-rwx "$l_file"
            fi
         fi
         if [ "$l_file_owner" != "root" ]; then
            a_out2+=("    Owned by: \"$l_file_owner\" should be owned by \"root\"" \
            "    Changing ownership to \"root\"")
            chown root "$l_file"
         fi
         if [[ ! "$l_file_group" =~ ($l_ssh_group_name|root) ]]; then
            [ -n "$l_ssh_group_name" ] && l_new_group="$l_ssh_group_name" ||
l_new_group="root"
            a_out2+=("    Owned by group \"$l_file_group\" should be group owned by:
\"$l_ssh_group_name\" or \"root\"" \
            "    Changing group ownership to \"$l_new_group\"")
            chgrp "$l_new_group" "$l_file"
         fi
         if [ "${#a_out2[@]}" -gt "0" ]; then
            a_output2+=("  - File: \"$l_file\"" "${a_out2[@]}")
         else
            a_output+=("  - File: \"$l_file\"" \
            "Correct: mode: \"$l_file_mode\", owner: \"$l_file_owner\", and group
owner: \"$l_file_group\" configured")
         fi
      done < <(stat -Lc '%#a:%U:%G' "$l_file")
   }
   while IFS= read -r -d $'\0' l_file; do
      if ssh-keygen -lf &>/dev/null "$l_file"; then
         file "$l_file" | grep -Piq -- '\bopenssh\h+([^#\n\r]+\h+)?private\h+key\b' &&
f_file_access_fix
      fi
   done < <(find -L /etc/ssh -xdev -type f -print0 2>/dev/null)
   if [ "${#a_output2[@]}" -le "0" ]; then
      printf '%s\n' "" " - No access changes required" ""
   else
      printf '%s\n' "" " - Remediation results:" "${a_output2[@]}" ""
   fi
}
```

References:

1. NIST SP 800-53 Rev. 5: AC-3. MP-2
2. RHEL 8 STIG Rule ID: SV-230542r858814
3. RHEL 8 STIG Rule ID: SV-230287r880714

Additional Information:

```
Red Hat Enterprise Linux 8 Security Technical Implementation Guide
Version 1, Release: 14 Benchmark Date: 24 APR 2024

Vul ID: V-230287
Rule ID: SV-230287r880714
STIG ID: RHEL-08-010490
Severity: CAT II
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.2.3 Ensure access to SSH public host key files is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

An SSH public key is one of two files used in SSH public key authentication. In this authentication method, a public key is a key that can be used for verifying digital signatures generated using a corresponding private key. Only a public key that corresponds to a private key will be able to authenticate successfully.

Rationale:

If a public host key file is modified by an unauthorized user, the SSH service may be compromised.

Audit:

Run the following script to verify SSH public host key files are mode `0644` or more restrictive, owned by the `root` user, and owned by the `root` group:

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=()
   l_pmask="0133"; l_maxperm="$( printf '%o' $(( 0777 & ~$l_pmask )) )"
   f_file_chk()
   {
      while IFS=: read -r l_file_mode l_file_owner l_file_group; do
         a_out2=()
         if [ $(( $l_file_mode & $l_pmask )) -gt 0 ]; then
            a_out2+=("    Mode: \"$l_file_mode\" should be mode:
\"$l_maxperm\" or more restrictive")
         fi
         if [ "$l_file_owner" != "root" ]; then
            a_out2+=("    Owned by: \"$l_file_owner\" should be owned by:
\"root\"")
         fi
         if [ "$l_file_group" != "root" ]; then
            a_out2+=("    Owned by group \"$l_file_group\" should be group
owned by group: \"root\"")
         fi
         if [ "${#a_out2[@]}" -gt "0" ]; then
            a_output2+=("  - File: \"$l_file\"" "${a_out2[@]}")
         else
            a_output+=("  - File: \"$l_file\"" \
            "    Correct: mode: \"$l_file_mode\", owner: \"$l_file_owner\"
and group owner: \"$l_file_group\" configured")
         fi
      done < <(stat -Lc '%#a:%U:%G' "$l_file")
   }
   while IFS= read -r -d $'\0' l_file; do
      if ssh-keygen -lf &>/dev/null "$l_file"; then
         file "$l_file" | grep -Piq --
'\bopenssh\h+([^#\n\r]+\h+)?public\h+key\b' && f_file_chk
      fi
   done < <(find -L /etc/ssh -xdev -type f -print0 2>/dev/null)
   if [ "${#a_output2[@]}" -le 0 ]; then
      [ "${#a_output[@]}" -le 0 ] && a_output+=("  - No openSSH public keys
found")
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
   fi
}
```

Remediation:

Run the following script to set mode, ownership, and group on the public SSH host key files:

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=()
   l_pmask="0133"; l_maxperm="$( printf '%o' $(( 0777 & ~$l_pmask )) )"
   f_file_access_fix()
   {
      while IFS=: read -r l_file_mode l_file_owner l_file_group; do
         a_out2=()
         [ $(( $l_file_mode & $l_pmask )) -gt 0 ] && \
            a_out2+=("    Mode: \"$l_file_mode\" should be mode:
\"$l_maxperm\" or more restrictive" \
            "    updating to mode: \"$l_maxperm\"") && chmod u-x,go-wx
"$l_file"
         [ "$l_file_owner" != "root" ] && \
            a_out2+=("    Owned by: \"$l_file_owner\" should be owned by
\"root\"" \
            "    Changing ownership to \"root\"") && chown root "$l_file"
         [ "$l_file_group" != "root" ] && \
            a_out2+=("    Owned by group \"$l_file_group\" should be group
owned by: \"root\"" \
            "    Changing group ownership to \"root\"") && chgrp root
"$l_file"
         if [ "${#a_out2[@]}" -gt "0" ]; then
            a_output2+=("  - File: \"$l_file\"" "${a_out2[@]}")
         else
            a_output+=("  - File: \"$l_file\"" \
            "    Correct: mode: \"$l_file_mode\", owner: \"$l_file_owner\",
and group owner: \"$l_file_group\" configured")
         fi
      done < <(stat -Lc '%#a:%U:%G' "$l_file")
   }
   while IFS= read -r -d $'\0' l_file; do
      if ssh-keygen -lf &>/dev/null "$l_file"; then
         file "$l_file" | grep -Piq --
'\bopenssh\h+([^#\n\r]+\h+)?public\h+key\b' && f_file_access_fix
      fi
   done < <(find -L /etc/ssh -xdev -type f -print0 2>/dev/null)
   if [ "${#a_output2[@]}" -le "0" ]; then
      printf '%s\n' "" " - No access changes required" ""
   else
      printf '%s\n' " - Remediation results:" "${a_output2[@]}" ""
   fi
}
```

Default Value:

644 0/root 0/root

References:

1. NIST SP 800-53 Rev. 5: AC-3 CM-6 b MP-2
2. NIST SP 800-53A :: CM-6.1 (iv)
3. RHEL 8 STIG Vul ID: V-230286
4. RHEL 8 STIG Rule ID: SV-230286r627750

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.2.4 Ensure sshd Ciphers are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

This variable limits the ciphers that SSH can use during communication.

Notes:

- Some organizations may have stricter requirements for approved ciphers.
- Ensure that ciphers used are in compliance with site policy.
- The only "strong" ciphers currently FIPS 140 compliant are:
  - aes256-gcm@openssh.com
  - aes128-gcm@openssh.com
  - aes256-ctr
  - aes192-ctr
  - aes128-ctr

Rationale:

Weak ciphers that are used for authentication to the cryptographic module cannot be relied upon to provide confidentiality or integrity, and system data may be compromised.

- The Triple DES ciphers, as used in SSH, have a birthday bound of approximately four billion blocks, which makes it easier for remote attackers to obtain clear text data via a birthday attack against a long-duration encrypted session, aka a "Sweet32" attack.
- Error handling in the SSH protocol; Client and Server, when using a block cipher algorithm in Cipher Block Chaining (CBC) mode, makes it easier for remote attackers to recover certain plain text data from an arbitrary block of cipher text in an SSH session via unknown vectors.

Audit:

Run the following command to verify none of the "weak" ciphers are being used:

```
# sshd -T | grep -Pi --
'^ciphers\h+\"?([^#\n\r]+,)?((3des|blowfish|cast128|aes(128|192|256))-
cbc|arcfour(128|256)?|rijndael-cbc@lysator\.liu\.se|chacha20-
poly1305@openssh\.com)\b'
```

- IF - a line is returned, review the list of ciphers. If the line includes `chacha20-poly1305@openssh.com`, review `CVE-2023-48795` and verify the system has been patched. No ciphers in the list below should be returned as they're considered "weak":

```
3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
```

Remediation:

Edit the /etc/ssh/sshd_config file and add/modify the `Ciphers` line to contain a comma separated list of the site unapproved (weak) Ciphers preceded with a `-` above any `Include` entries:
Example:

```
Ciphers -3des-cbc,aes128-cbc,aes192-cbc,aes256-cbc,chacha20-
poly1305@openssh.com
```

- IF - `CVE-2023-48795` has been addressed, and it meets local site policy, `chacha20-poly1305@openssh.com` may be removed from the list of excluded ciphers.
Note: First occurrence of an option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

Ciphers chacha20-poly1305@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com

References:

1. https://nvd.nist.gov/vuln/detail/CVE-2023-48795
2. https://nvd.nist.gov/vuln/detail/CVE-2019-1543
3. https://nvd.nist.gov/vuln/detail/CVE-2016-2183
4. https://nvd.nist.gov/vuln/detail/CVE-2008-5161
5. https://www.openssh.com/txt/cbc.adv
6. https://www.openssh.com/txt/cbc.adv
7. SSHD_CONFIG(5)
8. NIST SP 800-53 Rev. 5: SC-8

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

5.2.5 Ensure sshd KexAlgorithms is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Key exchange is any method in cryptography by which cryptographic keys are exchanged between two parties, allowing use of a cryptographic algorithm. If the sender and receiver wish to exchange encrypted messages, each must be equipped to encrypt messages to be sent and decrypt messages received

Notes:

- Kex algorithms have a higher preference the earlier they appear in the list
- Some organizations may have stricter requirements for approved Key exchange algorithms
- Ensure that Key exchange algorithms used are in compliance with site policy
- The only Key Exchange Algorithms currently FIPS 140 approved are:
  - ecdh-sha2-nistp256
  - ecdh-sha2-nistp384
  - ecdh-sha2-nistp521
  - diffie-hellman-group-exchange-sha256
  - diffie-hellman-group16-sha512
  - diffie-hellman-group18-sha512
  - diffie-hellman-group14-sha256

Rationale:

Key exchange methods that are considered weak should be removed. A key exchange method may be weak because too few bits are used, or the hashing algorithm is considered too weak. Using weak algorithms could expose connections to man-in-the-middle attacks

Audit:

Run the following command to verify none of the "weak" Key Exchange algorithms are being used:

```
# sshd -T | grep -Pi -- 'kexalgorithms\h+([^#\n\r]+,)?(diffie-hellman-group1-sha1|diffie-hellman-group14-sha1|diffie-hellman-group-exchange-sha1)\b'

Nothing should be returned
```

The following are considered "weak" Key Exchange Algorithms, and should not be used:

```
diffie-hellman-group1-sha1
diffie-hellman-group14-sha1
diffie-hellman-group-exchange-sha1
```

Remediation:

Edit the `/etc/ssh/sshd_config` file and add/modify the `KexAlgorithms` line to contain a comma separated list of the site unapproved (weak) KexAlgorithms preceded with a `-` above any `Include` entries:
Example:

```
KexAlgorithms -diffie-hellman-group1-sha1,diffie-hellman-group14-sha1,diffie-hellman-group-exchange-sha1
```

Note: First occurrence of an option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

KexAlgorithms sntrup761x25519-sha512@openssh.com,curve25519-sha256,curve25519-sha256@libssh.org,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256,diffie-hellman-group16-sha512,diffie-hellman-group18-sha512,diffie-hellman-group14-sha256

References:

1. https://ubuntu.com/server/docs/openssh-crypto-configuration
2. NIST SP 800-53 Rev. 5: SC-8
3. SSHD(8)
4. SSHD_CONFIG(5)

Additional Information:

The supported algorithms are:

```
curve25519-sha256
curve25519-sha256@libssh.org
diffie-hellman-group1-sha1
diffie-hellman-group14-sha1
diffie-hellman-group14-sha256
diffie-hellman-group16-sha512
diffie-hellman-group18-sha512
diffie-hellman-group-exchange-sha1
diffie-hellman-group-exchange-sha256
ecdh-sha2-nistp256
ecdh-sha2-nistp384
ecdh-sha2-nistp521
sntrup4591761x25519-sha512@tinyssh.org
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

5.2.6 Ensure sshd MACs are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

This variable limits the types of MAC algorithms that SSH can use during communication.

Notes:

- Some organizations may have stricter requirements for approved MACs.
- Ensure that MACs used are in compliance with site policy.
- The only "strong" MACs currently FIPS 140 approved are:
    - HMAC-SHA1
    - HMAC-SHA2-256
    - HMAC-SHA2-384
    - HMAC-SHA2-512

Rationale:

MD5 and 96-bit MAC algorithms are considered weak and have been shown to increase exploitability in SSH downgrade attacks. Weak algorithms continue to have a great deal of attention as a weak spot that can be exploited with expanded computing power. An attacker that breaks the algorithm could take advantage of a MiTM position to decrypt the SSH tunnel and capture credentials and information.

Audit:

Run the following command to verify none of the "weak" MACs are being used:

```
# sshd -T | grep -Pi -- 'macs\h+([^#\n\r]+,)?(hmac-md5|hmac-md5-96|hmac-
ripemd160|hmac-sha1-96|umac-64@openssh\.com|hmac-md5-etm@openssh\.com|hmac-
md5-96-etm@openssh\.com|hmac-ripemd160-etm@openssh\.com|hmac-sha1-96-
etm@openssh\.com|umac-64-etm@openssh\.com)\b'

Nothing should be returned
```

Note: Review CVE-2023-48795 and verify the system has been patched. If the system has not been patched, review the use of the Encrypt Then Mac (etm) MACs.
The following are considered "weak" MACs, and should not be used:

```
hmac-md5
hmac-md5-96
hmac-ripemd160
hmac-sha1-96
umac-64@openssh.com
hmac-md5-etm@openssh.com
hmac-md5-96-etm@openssh.com
hmac-ripemd160-etm@openssh.com
hmac-sha1-96-etm@openssh.com
umac-64-etm@openssh.com
```

Remediation:

Edit the `/etc/ssh/sshd_config` file and add/modify the `MACs` line to contain a comma separated list of the site unapproved (weak) MACs preceded with a `-` above any `Include` entries:
Example:

```
MACs -hmac-md5,hmac-md5-96,hmac-ripemd160,hmac-sha1-96,umac-
64@openssh.com,hmac-md5-etm@openssh.com,hmac-md5-96-etm@openssh.com,hmac-
ripemd160-etm@openssh.com,hmac-sha1-96-etm@openssh.com,umac-64-
etm@openssh.com
```

- IF - CVE-2023-48795 has not been reviewed and addressed, the following `etm` MACs should be added to the exclude list: hmac-sha1-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com
Note: First occurrence of an option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

MACs umac-64-etm@openssh.com,umac-128-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha1-etm@openssh.com,umac-64@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512,hmac-sha1

References:

1. https://nvd.nist.gov/vuln/detail/CVE-2023-48795
2. More information on SSH downgrade attacks can be found here: http://www.mitls.org/pages/attacks/SLOTH
3. SSHD_CONFIG(5)
4. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●
v7	16.5 Encrypt Transmittal of Username and Authentication Credentials Ensure that all account usernames and authentication credentials are transmitted across networks using encrypted channels.		●	●

5.2.7 Ensure sshd access is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

There are several options available to limit which users and group can access the system via SSH. It is recommended that at least one of the following options be leveraged:

- `AllowUsers`:
  - The `AllowUsers` variable gives the system administrator the option of allowing specific users to `ssh` into the system. The list consists of space separated user names. Numeric user IDs are not recognized with this variable. If a system administrator wants to restrict user access further by only allowing the allowed users to log in from a particular host, the entry can be specified in the form of user@host.
- `AllowGroups`:
  - The `AllowGroups` variable gives the system administrator the option of allowing specific groups of users to `ssh` into the system. The list consists of space separated group names. Numeric group IDs are not recognized with this variable.
- `DenyUsers`:
  - The `DenyUsers` variable gives the system administrator the option of denying specific users to `ssh` into the system. The list consists of space separated user names. Numeric user IDs are not recognized with this variable. If a system administrator wants to restrict user access further by specifically denying a user's access from a particular host, the entry can be specified in the form of user@host.
- `DenyGroups`:
  - The `DenyGroups` variable gives the system administrator the option of denying specific groups of users to `ssh` into the system. The list consists of space separated group names. Numeric group IDs are not recognized with this variable.

Rationale:

Restricting which users can remotely access the system via SSH will help ensure that only authorized users access the system.

Audit:

Run the following command and verify the output:

```
# sshd -T | grep -Pi -- '^\h*(allow|deny)(users|groups)\h+\H+'
```

Verify that the output matches at least one of the following lines:

```
allowusers <userlist>
-OR-
allowgroups <grouplist>
-OR-
denyusers <userlist>
-OR-
denygroups <grouplist>
```

Review the list(s) to ensure included users and/or groups follow local site policy
- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep -Pi --
'^\h*(allow|deny)(users|groups)\h+\H+'
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set one or more of the parameters above any `Include` and `Match` set statements as follows:

```
AllowUsers <userlist>
 - AND/OR -
AllowGroups <grouplist>
```

Note:

- First occurrence of an option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a `.conf` file in an `Include` directory.
- **Be advised** that these options are "ANDed" together. If both `AllowUsers` and `AllowGroups` are set, connections will be limited to the list of users that are also a member of an allowed group. It is recommended that only one be set for clarity and ease of administration.
- It is easier to manage an allow list than a deny list. In a deny list, you could potentially add a user or group and forget to add it to the deny list.

Default Value:

None

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5: AC-3. MP-2
3. SSHD(8)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities.	●	●	●

5.2.8 Ensure sshd Banner is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `Banner` parameter specifies a file whose contents must be sent to the remote user before authentication is permitted. By default, no banner is displayed.

Rationale:

Banners are used to warn connecting users of the particular site's policy regarding connection. Presenting a warning message prior to the normal user login may assist the prosecution of trespassers on the computer system.

Audit:

Run the following command to verify `Banner` is set:

```
# sshd -T | grep -Pi -- '^banner\h+\/\H+'
```

Example:

```
banner /etc/issue.net
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep -Pi -- '^banner\h+\/\H+'
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).
Run the following command and verify that the contents or the file being called by the `Banner` argument match site policy:

```
# [ -e "$(sshd -T | awk '$1 == "banner" {print $2}')" ] && cat "$(sshd -T |
awk '$1 == "banner" {print $2}')"
```

Run the following command and verify no results are returned:

```
# grep -Psi -- "(\\\v|\\\r|\\\m|\\\s|\b$(grep '^ID=' /etc/os-release | cut -
d= -f2 | sed -e 's/"//g')\b)" "$(sshd -T | awk '$1 == "banner" {print $2}')"
```

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `Banner` parameter above any `Include` and `Match` entries as follows:

```
Banner /etc/issue.net
```

Note: First occurrence of a option takes precedence, Match set statements withstanding. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location. Edit the file being called by the `Banner` argument with the appropriate contents according to your site policy, remove any instances of `\m` , `\r` , `\s` , `\v` or references to the `OS platform`
Example:

```
# printf '%s\n' "Authorized users only. All activity may be monitored and
reported." > "$(sshd -T | awk '$1 == "banner" {print $2}')"
```

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.9 Ensure sshd ClientAliveInterval and ClientAliveCountMax are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Note: To clarify, the two settings described below are only meant for idle connections from a protocol perspective and are not meant to check if the user is active or not. An idle user does not mean an idle connection. SSH does not and never had, intentionally, the capability to drop idle users. In SSH versions before `8.2p1` there was a bug that caused these values to behave in such a manner that they were abused to disconnect idle users. This bug has been resolved in `8.2p1` and thus it can no longer be abused disconnect idle users.

The two options `ClientAliveInterval` and `ClientAliveCountMax` control the timeout of SSH sessions. Taken directly from `man 5 sshd_config`:

- `ClientAliveInterval` Sets a timeout interval in seconds after which if no data has been received from the client, sshd(8) will send a message through the encrypted channel to request a response from the client. The default is 0, indicating that these messages will not be sent to the client.
- `ClientAliveCountMax` Sets the number of client alive messages which may be sent without sshd(8) receiving any messages back from the client. If this threshold is reached while client alive messages are being sent, sshd will disconnect the client, terminating the session. It is important to note that the use of client alive messages is very different from TCPKeepAlive. The client alive messages are sent through the encrypted channel and therefore will not be spoofable. The TCP keepalive option en-abled by TCPKeepAlive is spoofable. The client alive mechanism is valuable when the client or server depend on knowing when a connection has become unresponsive. The default value is 3. If ClientAliveInterval is set to 15, and ClientAliveCountMax is left at the default, unresponsive SSH clients will be disconnected after approximately 45 seconds. Setting a zero ClientAliveCountMax disables connection termination.

Rationale:

In order to prevent resource exhaustion, appropriate values should be set for both `ClientAliveInterval` and `ClientAliveCountMax`. Specifically, looking at the source code, `ClientAliveCountMax` must be greater than zero in order to utilize the ability of SSH to drop idle connections. If connections are allowed to stay open indefinitely, this can potentially be used as a DDOS attack or simple resource exhaustion could occur over unreliable networks.

The example set here is a 45 second timeout. Consult your site policy for network timeouts and apply as appropriate.

Audit:

Run the following command and verify `ClientAliveInterval` and `ClientAliveCountMax` are greater than zero:

```
# sshd -T | grep -Pi -- '(clientaliveinterval|clientalivecountmax)'
```

Example Output:

```
clientaliveinterval 15
clientalivecountmax 3
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep -Pi --
'(clientaliveinterval|clientalivecountmax)'
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `ClientAliveInterval` and `ClientAliveCountMax` parameters above any `Include` and `Match` entries according to site policy.
Example:

```
ClientAliveInterval 15
ClientAliveCountMax 3
```

Note: First occurrence of a option takes precedence, Match set statements withstanding. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

ClientAliveInterval 0

ClientAliveCountMax 3

References:

1. SSHD_CONFIG(5)
2. SSHD(8)
3. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

https://bugzilla.redhat.com/show_bug.cgi?id=1873547

https://github.com/openssh/openssh-portable/blob/V_8_9/serverloop.c#L137

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.10 Ensure sshd HostbasedAuthentication is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `HostbasedAuthentication` parameter specifies if authentication is allowed through trusted hosts via the user of `.rhosts`, or `/etc/hosts.equiv`, along with successful public key client host authentication.

Rationale:

Even though the `.rhosts` files are ineffective if support is disabled in `/etc/pam.conf`, disabling the ability to use `.rhosts` files in SSH provides an additional layer of protection.

Audit:

Run the following command to verify `HostbasedAuthentication` is set to `no`:

```
# sshd -T | grep hostbasedauthentication

hostbasedauthentication no
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep hostbasedauthentication
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `HostbasedAuthentication` parameter to `no` above any `Include` and `Match` entries as follows:

```
HostbasedAuthentication no
```

Note: First occurrence of a option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location.

Default Value:

HostbasedAuthentication no

References:

1. SSHD_CONFIG(5)
2. SSHD(8)
3. NIST SP 800-53 :: CM-6 b
4. NIST SP 800-53A :: CM-6.1 (iv)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.2.11 Ensure sshd IgnoreRhosts is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `IgnoreRhosts` parameter specifies that `.rhosts` and `.shosts` files will not be used in `RhostsRSAAuthentication` or `HostbasedAuthentication`.

Rationale:

Setting this parameter forces users to enter a password when authenticating with SSH.

Audit:

Run the following command to verify `IgnoreRhosts` is set to `yes`:

```
# sshd -T | grep ignorerhosts

ignorerhosts yes
```

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `IgnoreRhosts` parameter to `yes` above any `Include` entry as follows:

```
IgnoreRhosts yes
```

Note: First occurrence of a option takes precedence. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location.

Default Value:

IgnoreRhosts yes

References:

1. SSHD_CONFIG(5)
2. SSHD(8)
3. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.2.12 Ensure sshd LoginGraceTime is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `LoginGraceTime` parameter specifies the time allowed for successful authentication to the SSH server. The longer the Grace period is the more open unauthenticated connections can exist. Like other session controls in this session the Grace Period should be limited to appropriate organizational limits to ensure the service is available for needed access.

Rationale:

Setting the `LoginGraceTime` parameter to a low number will minimize the risk of successful brute force attacks to the SSH server. It will also limit the number of concurrent unauthenticated connections While the recommended setting is 60 seconds (1 Minute), set the number based on site policy.

Audit:

Run the following command and verify that output `LoginGraceTime` is between `1` and `60` seconds:

```
# sshd -T | grep logingracetime

logingracetime 60
```

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `LoginGraceTime` parameter to `60` seconds or less above any `Include` entry as follows:

```
LoginGraceTime 60
```

Note: First occurrence of a option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

LoginGraceTime 120

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5: CM-6
3. SSHD(8)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.13 Ensure sshd LogLevel is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

SSH provides several logging levels with varying amounts of verbosity. The `DEBUG` options are specifically not recommended other than strictly for debugging SSH communications. These levels provide so much data that it is difficult to identify important security information, and may violate the privacy of users.

Rationale:

The `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.

The `VERBOSE` level specifies that login and logout activity as well as the key fingerprint for any SSH key used for login will be logged. This information is important for SSH key management, especially in legacy environments.

Audit:

Run the following command and verify that output matches `loglevel VERBOSE` or `loglevel INFO`:

```
# sshd -T | grep loglevel

loglevel VERBOSE
   - OR -
loglevel INFO
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep loglevel
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `LogLevel` parameter to `VERBOSE` or `INFO` above any `Include` and `Match` entries as follows:

```
LogLevel VERBOSE
   - OR -
LogLevel INFO
```

Note: First occurrence of an option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location.

Default Value:

LogLevel INFO

References:

1. https://www.ssh.com/ssh/sshd_config/
2. NIST SP 800-53 Rev. 5: AU-3, AU-12, SI-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

5.2.14 Ensure sshd MaxAuthTries is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `MaxAuthTries` parameter specifies the maximum number of authentication attempts permitted per connection. When the login failure count reaches half the number, error messages will be written to the `syslog` file detailing the login failure.

Rationale:

Setting the `MaxAuthTries` parameter to a low number will minimize the risk of successful brute force attacks to the SSH server. While the recommended setting is 4, set the number based on site policy.

Audit:

Run the following command and verify that `MaxAuthTries` is `4` or less:

```
# sshd -T | grep maxauthtries

maxauthtries 4
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep maxauthtries
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `MaxAuthTries` parameter to `4` or less above any `Include` and `Match` entries as follows:

```
MaxAuthTries 4
```

Note: First occurrence of an option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location.

Default Value:

MaxAuthTries 6

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5: AU-3

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.5 Collect Detailed Audit Logs Configure detailed audit logging for enterprise assets containing sensitive data. Include event source, date, username, timestamp, source addresses, destination addresses, and other useful elements that could assist in a forensic investigation.		●	●
v7	16.13 Alert on Account Login Behavior Deviation Alert when users deviate from normal login behavior, such as time-of-day, workstation location and duration.			●

5.2.15 Ensure sshd MaxStartups is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `MaxStartups` parameter specifies the maximum number of concurrent unauthenticated connections to the SSH daemon.

Rationale:

To protect a system from denial of service due to a large number of pending authentication connection attempts, use the rate limiting function of MaxStartups to protect availability of sshd logins and prevent overwhelming the daemon.

Audit:

Run the following command to verify `MaxStartups` is `10:30:100` or more restrictive:

```
# sshd -T | awk '$1 ~ /^\s*maxstartups/{split($2, a, ":");{if(a[1] > 10 ||
a[2] > 30 || a[3] > 100) print $0}}'
```

Nothing should be returned

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `MaxStartups` parameter to `10:30:100` or more restrictive above any `Include` entries as follows:

```
MaxStartups 10:30:100
```

Note: First occurrence of a option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

MaxStartups 10:30:100

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.16 Ensure sshd MaxSessions is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `MaxSessions` parameter specifies the maximum number of open sessions permitted from a given connection.

Rationale:

To protect a system from denial of service due to a large number of concurrent sessions, use the rate limiting function of MaxSessions to protect availability of sshd logins and prevent overwhelming the daemon.

Audit:

Run the following command and verify that `MaxSessions` is `10` or less:

```
# sshd -T | grep -i maxsessions

maxsessions 10
```

Run the following command and verify the output:

```
grep -Psi -- '^\h*MaxSessions\h+\"?(1[1-9]|[2-9][0-9]|[1-9][0-9][0-9]+)\b'
/etc/ssh/sshd_config /etc/ssh/sshd_config.d/*.conf

Nothing should be returned.
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep maxsessions
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `MaxSessions` parameter to `10` or less above any `Include` and `Match` entries as follows:

```
MaxSessions 10
```

Note: First occurrence of an option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location.

Default Value:

MaxSessions 10

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.17 Ensure sshd PermitEmptyPasswords is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `PermitEmptyPasswords` parameter specifies if the SSH server allows login to accounts with empty password strings.

Rationale:

Disallowing remote shell access to accounts that have an empty password reduces the probability of unauthorized access to the system.

Audit:

Run the following command to verify `PermitEmptyPasswords` is set to `no`:

```
# sshd -T | grep permitemptypasswords

permitemptypasswords no
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep permitemptypasswords
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit `/etc/ssh/sshd_config` and set the `PermitEmptyPasswords` parameter to `no` above any `Include` and `Match` entries as follows:

```
PermitEmptyPasswords no
```

Note: First occurrence of an option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location. The SSH daemon must be restarted for the changes to take effect. To restart the SSH daemon, run the following command:

```
# systemctl reload-or-restart sshd.service
```

Default Value:

PermitEmptyPasswords no

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Revision 5 :: CM-6 b
3. NIST SP 800-53A :: CM-6.1 (iv)
4. RHEL 8 STIG Vul ID: V-230380
5. RHEL 8 STIG Rule ID: SV-230380r951612

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.2.18 Ensure sshd PermitRootLogin is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `PermitRootLogin` parameter specifies if the root user can log in using SSH. The default is `prohibit-password`.

Rationale:

Disallowing `root` logins over SSH requires system admins to authenticate using their own individual account, then escalating to `root`. This limits opportunity for non-repudiation and provides a clear audit trail in the event of a security incident.

Audit:

Run the following command to verify `PermitRootLogin` is set to `no`:

```
# sshd -T | grep permitrootlogin

permitrootlogin no
```

- IF - `Match` set statements are used in your environment, specify the connection parameters to use for the `-T` extended test mode and run the audit to verify the setting is not incorrectly configured in a match block.
Example additional audit needed for a match block for the user `sshuser`:

```
# sshd -T -C user=sshuser | grep permitrootlogin
```

Note: If provided, any Match directives in the configuration file that would apply are applied before the configuration is written to standard output. The connection parameters are supplied as keyword=value pairs and may be supplied in any order, either with multiple `-C` options or as a comma-separated list. The keywords are `addr` (source address), `user` (user), `host` (resolved source host name), `laddr` (local address), `lport` (local port number), and `rdomain` (routing domain).

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `PermitRootLogin` parameter to `no` above any `Include` and `Match` entries as follows:

```
PermitRootLogin no
```

Note: First occurrence of an option takes precedence, `Match` set statements withstanding. If `Include` locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in `Include` location.

Default Value:

PermitRootLogin without-password

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5:AC-6
3. NIST SP 800-53A :: IA-2 (5).2 (ii)
4. RHEL 8 STIG Vul ID: V-230296
5. RHEL 8 STIG Rule ID: SV-230296r951608

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account.	●	●	●
v7	4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities.	●	●	●

5.2.19 Ensure sshd PermitUserEnvironment is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `PermitUserEnvironment` option allows users to present environment options to the SSH daemon.

Rationale:

Permitting users the ability to set environment variables through the SSH daemon could potentially allow users to bypass security controls (e.g. setting an execution path that has SSH executing trojan'd programs).

Audit:

Run the following command to verify `PermitUserEnvironment` is set to `no`:

```
# sshd -T | grep permituserenvironment

permituserenvironment no
```

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `PermitUserEnvironment` parameter to `no` above any `Include` entries as follows:

```
PermitUserEnvironment no
```

Note: First occurrence of an option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

PermitUserEnvironment no

References:

1. SSHD_CONFIG(5)
2. SSHD(8)
3. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5
4. NIST SP 800-53A :: CM-6.1 (iv)
5. RHEL 8 STIG Vul ID: V-230330
6. RHEL 8 STIG Rule ID: SV-230330r951610

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.20 Ensure sshd UsePAM is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `UsePAM` directive enables the Pluggable Authentication Module (PAM) interface. If set to `yes` this will enable PAM authentication using `ChallengeResponseAuthentication` and `PasswordAuthentication` directives in addition to PAM account and session module processing for all authentication types.

Rationale:

When `usePAM` is set to `yes`, PAM runs through account and session types properly. This is important if you want to restrict access to services based off of IP, time or other factors of the account. Additionally, you can make sure users inherit certain environment variables on login or disallow access to the server.

Audit:

Run the following command to verify `UsePAM` is set to `yes`:

```
# sshd -T | grep -i usepam

usepam yes
```

Remediation:

Edit the `/etc/ssh/sshd_config` file to set the `UsePAM` parameter to `yes` above any `Include` entries as follows:

```
UsePAM yes
```

Note: First occurrence of an option takes precedence. If Include locations are enabled, used, and order of precedence is understood in your environment, the entry may be created in a file in Include location.

Default Value:

UsePAM yes

References:

1. SSHD_CONFIG(5)
2. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5
3. SSHD(8)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.3 Configure privilege escalation

There are various tools which allows a permitted user to execute a command as the superuser or another user, as specified by the security policy.

sudo

https://www.sudo.ws/

The invoking user's real (not effective) user ID is used to determine the user name with which to query the security policy.

`sudo` supports a plug-in architecture for security policies and input/output logging. Third parties can develop and distribute their own policy and I/O logging plug-ins to work seamlessly with the `sudo` front end. The default security policy is `sudoers`, which is configured via the file `/etc/sudoers` and any entries in `/etc/sudoers.d`.

pkexec

https://www.freedesktop.org/software/polkit/docs/0.105/pkexec.1.html

5.3.1 Ensure sudo is installed (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`sudo` allows a permitted user to execute a command as the superuser or another user, as specified by the security policy. The invoking user's real (not effective) user ID is used to determine the user name with which to query the security policy.

Rationale:

`sudo` supports a plug-in architecture for security policies and input/output logging. Third parties can develop and distribute their own policy and I/O logging plug-ins to work seamlessly with the `sudo` front end. The default security policy is `sudoers`, which is configured via the file `/etc/sudoers` and any entries in `/etc/sudoers.d`.

The security policy determines what privileges, if any, a user has to run `sudo`. The policy may require that users authenticate themselves with a password or another authentication mechanism. If authentication is required, `sudo` will exit if the user's password is not entered within a configurable time limit. This limit is policy-specific.

Audit:

Verify that `sudo` is installed.
Run the following command:

```
# tdnf list sudo

Installed Packages
sudo.x86_64            <VERSION>        @anaconda
Available Packages
sudo.x86_64            <VERSION>        updates
```

Remediation:

Run the following command to install sudo

```
# dnf install sudo
```

References:

1. SUDO(8)
2. NIST SP 800-53 Rev. 5: AC-6

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account.	●	●	●
v7	4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities.	●	●	●

5.3.2 Ensure re-authentication for privilege escalation is not disabled globally (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The operating system must be configured so that users must re-authenticate for privilege escalation.

Rationale:

Without re-authentication, users may access resources or perform tasks for which they do not have authorization.

When operating systems provide the capability to escalate a functional capability, it is critical the user re-authenticate.

Audit:

Verify the operating system requires users to re-authenticate for privilege escalation. Check the configuration of the `/etc/sudoers` and `/etc/sudoers.d/*` files with the following command:

```
# grep -r "^[^#].*\!authenticate" /etc/sudoers*
```

If any line is found with a `!authenticate` tag, refer to the remediation procedure below.

Remediation:

Configure the operating system to require users to reauthenticate for privilege escalation.
Based on the outcome of the audit procedure, use `visudo -f <PATH TO FILE>` to edit the relevant sudoers file.
Remove any occurrences of `!authenticate` tags in the file(s).

References:

1. NIST SP 800-53 Rev. 5: AC-6

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account.	●	●	●
v7	4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities.	●	●	●

5.3.3 Ensure sudo authentication timeout is configured correctly (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`sudo` caches used credentials for a default of 5 minutes. This is for ease of use when there are multiple administrative tasks to perform. The timeout can be modified to suit local security policies.

Rationale:

Setting a timeout value reduces the window of opportunity for unauthorized privileged access to another user.

Audit:

Ensure that the caching timeout is no more than 15 minutes.
Example:

```
# grep -roP "timestamp_timeout=\K[0-9]*" /etc/sudoers*
```

If there is no `timestamp_timeout` configured in `/etc/sudoers*` then the default is 5 minutes. This default can be checked with:

```
# sudo -V | grep "Authentication timestamp timeout:"
```

NOTE: A value of `-1` means that the timeout is disabled. Depending on the configuration of the `timestamp_type`, this could mean for all terminals / processes of that user and not just that one single terminal session.

Remediation:

If the currently configured timeout is larger than 15 minutes, edit the file listed in the audit section with `visudo -f <PATH TO FILE>` and modify the entry `timestamp_timeout=` to 15 minutes or less as per your site policy. The value is in minutes. This particular entry may appear on its own, or on the same line as `env_reset`. See the following two examples:

```
Defaults    env_reset, timestamp_timeout=15
Defaults    timestamp_timeout=15
Defaults    env_reset
```

References:

1. https://www.sudo.ws/man/1.9.0/sudoers.man.html
2. NIST SP 800-53 Rev. 5: AC-6

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account.	●	●	●
v7	4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities.	●	●	●

5.4 Configure PAM

PAM (Pluggable Authentication Modules) is a service that implements modular authentication modules on UNIX systems. PAM is implemented as a set of shared objects that are loaded and executed when a program needs to authenticate a user. Files for PAM are typically located in the `/etc/pam.d` directory. PAM must be carefully configured to secure system authentication. While this section covers some of PAM, please consult other PAM resources to fully understand the configuration capabilities.

5.4.1 Ensure password creation requirements are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `pam_pwquality.so` module checks the strength of passwords. It performs checks such as making sure a password is not a dictionary word, it is a certain length, contains a mix of characters (e.g. alphabet, numeric, other) and more. The following are definitions of the `pam_pwquality.so` options.

The following options are set in the `/etc/security/pwquality.conf` file:

Password Length:

- `minlen = 14` - password must be 14 characters or more

Password complexity:

- `minclass = 4` - The minimum number of required classes of characters for the new password (digits, uppercase, lowercase, others)

  -OR-

- `dcredit = -1` - provide at least one digit
- `ucredit = -1` - provide at least one uppercase character
- `ocredit = -1` - provide at least one special character
- `lcredit = -1` - provide at least one lowercase character

The following is set in the `/etc/pam.d/system-password` and `/etc/pam.d/system-auth` files:

- `try_first_pass` - retrieve the password from a previous stacked PAM module. If not available, then prompt the user for a password.
- `retry=3` - Allow 3 tries before sending back a failure.

The settings shown above are one possible policy. Alter these values to conform to your own organization's password policies.

Notes:

- Settings in `/etc/security/pwquality.conf` must use spaces around the `=` symbol.
- Additional modules options may be set in the `/etc/pam.d/system-password` and `/etc/pam.d/system-auth` files.

Rationale:

Strong passwords and limited attempts before locking an account protect systems from being hacked through brute force methods.

Audit:

Verify password creation requirements conform to organization policy.
Run the following command to verify the minimum password length is 14 or more characters:

```
# grep -Pi '^\h*minlen\b' /etc/security/pwquality.conf

minlen = 14
```

Run one of the following commands to verify the required password complexity:

```
# grep -Pi '^\h*minclass\b' /etc/security/pwquality.conf

minclass = 4
```

-OR-

```
# grep -Pi '^\h*[duol]credit\b' /etc/security/pwquality.conf

dcredit = -1
ucredit = -1
lcredit = -1
ocredit = -1
```

Run the following commands to verify the files: `/etc/pam.d/system-password` and `/etc/pam.d/system-auth` include `retry=3` on the `password requisite pam_pwquality.so` line:

```
# grep -P
'^\h*password\h+([^#\n\r]+\h+)?pam_pwquality\.so\h+([^#\n\r]+\h+)?(retry=[1-
3])\b' /etc/pam.d/system-password
```

Example output:

```
password    requisite     pam_pwquality.so retry=3
# grep -P
'^\h*password\h+([^#\n\r]+\h+)?pam_pwquality\.so\h+([^#\n\r]+\h+)?(retry=[1-
3])\b' /etc/pam.d/system-auth
```

Example output:

```
password    requisite     pam_pwquality.so retry=3
```

Remediation:

Edit the file `/etc/security/pwquality.conf` and add or modify the following line for password length to conform to site policy:

```
minlen = 14
```

Edit the file `/etc/security/pwquality.conf` and add or modify the following line for password complexity to conform to site policy:

```
minclass = 4
```

-OR-

```
dcredit = -1
ucredit = -1
ocredit = -1
lcredit = -1
```

Edit the `/etc/pam.d/system-password` and `/etc/pam.d/system-auth` files to include the appropriate options for `pam_pwquality.so` and to conform to site policy:

```
password requisite pam_pwquality.so retry=3
```

References:

1. NIST SP 800-53 Rev. 5: IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.4.2 Ensure lockout for failed password attempts is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Lock out users after *n* unsuccessful consecutive login attempts.

- `deny=<n>` - Number of attempts before the account is locked
- `unlock_time=<n>` - Time in seconds before the account is unlocked

Note: The maximum configurable value for `unlock_time` is `604800`.

Rationale:

Locking out user IDs after *n* unsuccessful consecutive login attempts mitigates brute force password attacks against your systems.

Audit:

Verify password lockouts are configured. Depending on the version you are running, follow **one** of the two methods below.

- `deny` **should not** be `0` (never) or greater than `5`.
- `unlock_time` **should** be `0` (never) or `900` seconds or more.

These settings are commonly configured with the `pam_failock.so` module found in `/etc/pam.d/system-auth` and `/etc/pam.d/system-password`.
Run the following command and review the output to ensure that it follows local site policy.

```
# grep -P '^\h*auth\h+[^#\n\r]+\h+pam_faillock.so\s+' /etc/pam.d/system-
password /etc/pam.d/system-auth
```

Output should look similar to:

```
/etc/pam.d/system-password:auth     required     pam_faillock.so preauth
silent deny=5 unlock_time=900
/etc/pam.d/system-password:auth     required     pam_faillock.so authfail
deny=5 unlock_time=900
/etc/pam.d/system-auth:auth       required     pam_faillock.so preauth silent
deny=5 unlock_time=900
/etc/pam.d/system-auth:auth       required     pam_faillock.so authfail
deny=5 unlock_time=900
```

Remediation:

Set password lockouts and unlock times to conform to site policy. deny should be not greater than `5` and unlock_time should be `0` (never), or `900` seconds or greater.
Edit the files `/etc/pam.d/system-auth` and `/etc/pam.d/system-password` and add the following lines:
Modify the `deny=` and `unlock_time=` parameters to conform to local site policy, Not to be greater than `deny=5`:
Add the following lines to the `auth` section:

```
auth        required      pam_faillock.so preauth silent audit deny=5
unlock_time=900
auth        [default=die] pam_faillock.so authfail audit deny=5
unlock_time=900
```

The `auth` sections should look similar to the following example:
Note: The ordering on the lines in the auth section is important. The `preauth` line needs to below the line `auth required pam_env.so` and above all password validation lines. The `authfail` line needs to be after all password validation lines such as `pam_sss.so`. Incorrect order can cause you to be locked out of the system
Example:

```
auth        required      pam_env.so
auth        required      pam_faillock.so preauth silent audit deny=5
unlock_time=900 # <- Under "auth required pam_env.so"
auth        sufficient    pam_unix.so nullok try_first_pass
auth        [default=die] pam_faillock.so authfail audit deny=5
unlock_time=900 # <- Last auth line before "auth requisite
pam_succeed_if.so"
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        required      pam_deny.so
```

Add the following line to the `account` section:

```
account     required      pam_faillock.so
```

Example:

```
account     required     pam_faillock.so
account     required     pam_unix.so
account     sufficient   pam_localuser.so
account     sufficient   pam_pam_succeed_if.so uid < 1000 quiet
account     required     pam_permit.so
```

References:

1. https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/security_guide
2. NIST SP 800-53 Rev. 5: AC-1, AC-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.1 Establish and Maintain a Secure Configuration Process Establish and maintain a secure configuration process for enterprise assets (end-user devices, including portable and mobile, non-computing/IoT devices, and servers) and software (operating systems and applications). Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	16.7 Establish Process for Revoking Access Establish and follow an automated process for revoking system access by disabling accounts immediately upon termination or change of responsibilities of an employee or contractor . Disabling these accounts, instead of deleting accounts, allows preservation of audit trails.		●	●

5.4.3 Ensure password hashing algorithm is SHA-512 (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The commands below change password encryption from `md5` to `sha512` (a much stronger hashing algorithm). All existing accounts will need to perform a password change to upgrade the stored hashes to the new algorithm.

Note:

- These changes only apply to accounts configured on the local system.
- Additional module options may be set, recommendation only covers those listed here.

Rationale:

The SHA-512 algorithm provides much stronger hashing than MD5, thus providing additional protection to the system by increasing the level of effort for an attacker to successfully determine passwords.

Audit:

Run the following command to verify the sha512 option is included:

```
# grep -P
'^\h*password\h+([^#\n\r]+)?\h+pam_unix\.so\h+([^#\n\r]+\h+)?sha512\b'
/etc/pam.d/system-auth /etc/pam.d/system-password
```

Output should be similar to:

```
/etc/pam.d/system-auth:password    sufficient    pam_unix.so sha512 shadow
try_first_pass use_authtok
/etc/pam.d/system-password:password    sufficient    pam_unix.so sha512
shadow try_first_pass use_authtok
```

Remediation:

Edit the `/etc/pam.d/system-password` and `/etc/pam.d/system-auth` files to include `sha512` option and remove the `md5` option for `pam_unix.so`:

```
password sufficient pam_unix.so sha512
```

Note:

- Any system accounts that need to be expired should be carefully done separately by the system administrator to prevent any potential problems.
- If it is determined that the password algorithm being used is not SHA-512, once it is changed, it is recommended that all user ID's be immediately expired and forced to change their passwords on next login, In accordance with local site policies.

Default Value:

sha512

References:

1. NIST SP 800-53 Rev. 5: IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	16.4 Encrypt or Hash all Authentication Credentials Encrypt or hash with a salt all authentication credentials when stored.		●	●

5.4.4 Ensure password reuse is limited (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/security/opasswd` file stores the users' old passwords and can be checked to ensure that users are not recycling recent passwords.

Rationale:

Forcing users not to reuse their past 5 passwords makes it less likely that an attacker will be able to guess the password.

Audit:

Verify remembered password history follows local site policy, not to be less than `5`.
Run the following command:

```
# grep -P
'^\h*password\h+[^#\n\r]+\h+pam_pwhistory\.so\h+([^#\n\r]+\h+)?remember=([5-
9]|[1-9][0-9]+)\b' /etc/pam.d/system-password /etc/pam.d/system-auth
```

Output should look similar to:

```
/etc/pam.d/system-auth:password    requisite      pam_pwhistory.so remember=5
/etc/pam.d/system-password:password    requisite      pam_pwhistory.so
remember=5
```

Remediation:

Edit **both** the `/etc/pam.d/system-password` and `/etc/pam.d/system-auth` files to include the remember option and conform to site policy as shown:
Note: Add or modify the line containing the `pam_pwhistory.so` **after the first occurrence of `password requisite`:**

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

Example: (**Second line is modified**)

```
password    requisite     pam_pwquality.so try_first_pass local_users_only
authtok_type=
password    requisite     pam_pwhistory.so use_authtok remember=5 retry=3
password    sufficient    pam_unix.so sha512 shadow try_first_pass
use_authtok
password    required      pam_deny.so
```

Additional Information:

- This setting only applies to local accounts.
- This option is configured with the remember=*n* module option in `/etc/pam.d/system-auth` and `/etc/pam.d/system-password`

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	16 Account Monitoring and Control Account Monitoring and Control			

5.5 User Accounts and Environment

This section provides guidance on setting up secure defaults for system and user accounts and their environment.

5.5.1 Set Shadow Password Suite Parameters

While a majority of the password control parameters have been moved to PAM, some parameters are still available through the shadow password suite. Any changes made to `/etc/login.defs` will only be applied if the `usermod` command is used. If user IDs are added a different way, use the `chage` command to effect changes to individual user IDs.

5.5.1.1 Ensure password expiration is 365 days or less (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The PASS_MAX_DAYS parameter in /etc/login.defs allows an administrator to force passwords to expire once they reach a defined age. It is recommended that the PASS_MAX_DAYS parameter be set to less than or equal to 365 days.

Rationale:

The window of opportunity for an attacker to leverage compromised credentials or successfully compromise credentials via an online brute force attack is limited by the age of the password. Therefore, reducing the maximum age of a password also reduces an attacker's window of opportunity.

Audit:

Run the following command and verify PASS_MAX_DAYS conforms to site policy (no more than 365 days):

```
# grep PASS_MAX_DAYS /etc/login.defs

PASS_MAX_DAYS 365
```

Run the following command and review list of users and PASS_MAX_DAYS to verify that all users' PASS_MAX_DAYS conforms to site policy (no more than 365 days):

```
# awk -F: '$2~/^[^*!xX\n\r][^\n\r]+/{print $1":"$5}' /etc/shadow

<user>:<PASS_MAX_DAYS>
```

Remediation:

Set the PASS_MAX_DAYS parameter to conform to site policy in /etc/login.defs:

```
PASS_MAX_DAYS 365
```

Modify user parameters for all users with a password set to match:

```
# chage --maxdays 365 <user>
```

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

You can also check this setting in `/etc/shadow` directly. The 5th field should be 365 or less for all users with a password.

Note: A value of -1 will disable password expiration. Additionally the password expiration must be greater than the minimum days between password changes or users will be unable to change their password.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.5.1.2 Ensure minimum days between password changes is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The PASS_MIN_DAYS parameter in /etc/login.defs allows an administrator to prevent users from changing their password until a minimum number of days have passed since the last time the user changed their password. It is recommended that PASS_MIN_DAYS parameter be set to 1 or more days.

Rationale:

By restricting the frequency of password changes, an administrator can prevent users from repeatedly changing their password in an attempt to circumvent password reuse controls.

Audit:

Run the following command and verify PASS_MIN_DAYS conforms to site policy (no less than 1 day):

```
# grep ^\s*PASS_MIN_DAYS /etc/login.defs

PASS_MIN_DAYS 1
```

Run the following command and review list of users and PASS_MIN_DAYS to verify that all users' PASS_MIN_DAYS conforms to site policy (no less than 1 day):

```
# awk -F: '$2~/^[^*!xX\n\r][^\n\r]+/{print $1":"$4}' /etc/shadow

<user>:<PASS_MIN_DAYS>
```

Remediation:

Set the PASS_MIN_DAYS parameter to 1 in /etc/login.defs:

```
PASS_MIN_DAYS 1
```

Modify user parameters for all users with a password set to match:

```
# chage --mindays 1 <user>
```

Default Value:

PASS_MIN_DAYS 0

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

You can also check this setting in `/etc/shadow` directly. The 4th field should be 1 or more for all users with a password.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.5.1.3 Ensure password expiration warning days is 7 or more (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The PASS_WARN_AGE parameter in /etc/login.defs allows an administrator to notify users that their password will expire in a defined number of days. It is recommended that the PASS_WARN_AGE parameter be set to 7 or more days.

Rationale:

Providing an advance warning that a password will be expiring gives users time to think of a secure password. Users caught unaware may choose a simple password or write it down where it may be discovered.

Audit:

Run the following command and verify PASS_WARN_AGE conforms to site policy (No less than 7 days):

```
# grep PASS_WARN_AGE /etc/login.defs

PASS_WARN_AGE 7
```

Verify all users with a password have their number of days of warning before password expires set to 7 or more. Run the following command and review list of users and PASS_WARN_AGE to verify that all users' PASS_WARN_AGE conforms to site policy (No less than 7 days):

```
# awk -F: '$2~/[^*!xX\n\r][^\n\r]+/{print $1":"$6}' /etc/shadow

<user>:<PASS_WARN_AGE>
```

Remediation:

Set the PASS_WARN_AGE parameter to 7 in /etc/login.defs:

```
PASS_WARN_AGE 7
```

Modify user parameters for all users with a password set to match:

```
# chage --warndays 7 <user>
```

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

You can also check this setting in `/etc/shadow` directly. The 6th field should be 7 or more for all users with a password.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.5.1.4 Ensure inactive password lock is 30 days or less (Automated)

Profile Applicability:

- Level 1 - Server

Description:

User accounts that have been inactive for over a given period of time can be automatically disabled. It is recommended that accounts that are inactive for 30 days after password expiration be disabled.

Rationale:

Inactive accounts pose a threat to system security since the users are not logging in to notice failed login attempts or other anomalies.

Audit:

Run the following command and verify `INACTIVE` conforms to site policy (no more than 30 days):

```
# useradd -D | grep INACTIVE

INACTIVE=30
```

Run the following command and review list of users and INACTIVE to verify that all users' INACTIVE conforms to site policy (no more than 30 days):

```
# awk -F: '$2~/^[^*!xX\n\r][^\n\r]+/{print $1":"$7}' /etc/shadow

<user>:<INACTIVE>
```

Remediation:

Run the following command to set the default password inactivity period to 30 days:

```
# useradd -D -f 30
```

Modify user parameters for all users with a password set to match:

```
# chage --inactive 30 <user>
```

Default Value:

INACTIVE=-1

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

You can also check this setting in `/etc/shadow` directly. The 7th field should be 30 or less for all users with a password.

Note: A value of -1 would disable this setting.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.5.1.5 Ensure all users last password change date is in the past (Automated)

Profile Applicability:

- Level 1 - Server

Description:

All users should have a password change date in the past.

Rationale:

If a users recorded password change date is in the future then they could bypass any set password expiration.

Audit:

Run the following command and verify nothing is returned:

```
{
   l_output2=""
   while read -r l_user; do
      l_change="$(chage --list $l_user | awk -F: '($1 ~
/^\s*Last\s+password\s+change/ && $2 !~ /never/){print $2}' | xargs)"
      if [[ "$(date -d "$l_change" +%s)" -gt "$(date +%s)" ]]; then
         l_output2="$l_output2\n  - User: \"$l_user\" last password change is
in the future \"$l_change\""
      fi
   done < <(awk -F: '($2 ~ /^[^*!xX\n\r][^\n\r]+/){print $1}' /etc/shadow)
   if [ -z "$l_output2" ]; then # If l_output2 is empty, we pass
      echo -e "\n- Audit Result:\n  ** PASS **\n - All user password changes
are in the past \n"
   else
      echo -e "\n- Audit Result:\n  ** FAIL **\n - * Reasons for audit
failure * :$l_output2\n"
   fi
}
```

Remediation:

Investigate any users with a password change date in the future and correct them. Locking the account, expiring the password, or resetting the password manually may be appropriate.

References:

1. NIST SP 800-53 Rev. 5: IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

5.5.2 Ensure system accounts are secured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

There are a number of accounts provided with most distributions that are used to manage applications and are not intended to provide an interactive shell.

Rationale:

It is important to make sure that accounts that are not being used by regular users are prevented from being used to provide an interactive shell. By default, most distributions set the password field for these accounts to an invalid string, but it is also recommended that the shell field in the password file be set to the `nologin` shell. This prevents the account from potentially being used to run any commands.

Audit:

Run the following script to verify all local system accounts:

- Do not have a valid login shell
- Are locked

```
#!/usr/bin/env bash

{
   l_output="" l_output2=""
   l_valid_shells="^($( awk -F\/ '$NF != "nologin" {print}' /etc/shells | sed
-rn '/^\//{s,/,\\\\/,g;p}' | paste -s -d '|' - ))$"
   a_users=(); a_ulock=() # initialize arrays
   while read -r l_user; do # Populate array with system accounts that have a
valid login shell
      a_users+=("$l_user")
   done < <(awk -v pat="$l_valid_shells" -F:
'($1!~/(root|sync|shutdown|halt|^\+)/ && $3<'"$(awk '/^\s*UID_MIN/{print $2}'
/etc/login.defs)"' && $(NF) ~ pat) { print $1 }' /etc/passwd)
   while read -r l_ulock; do # Populate array with system accounts that
aren't locked
      a_ulock+=("$l_ulock")
   done < <(awk -v pat="$l_valid_shells" -F: '($1!~/(root|^\+)/ && $2!~/LK?/
&& $3<'"$(awk '/^\s*UID_MIN/{print $2}' /etc/login.defs)"' && $(NF) ~ pat) {
print $1 }' /etc/passwd)
   if ! (( ${#a_users[@]} > 0 )); then
      l_output="$l_output\n  - local system accounts login is disabled"
   else
      l_output2="$l_output2\n  - There are \"$(printf '%s'
"${#a_users[@]}")\" system accounts with login enabled\n   - List of
accounts:\n$(printf '%s\n' "${a_users[@]:0:$l_limit}")\n   - end of list\n"
   fi
   if ! (( ${#a_ulock[@]} > 0 )); then
      l_output="$l_output\n  - local system accounts are locked"
   else
      l_output2="$l_output2\n  - There are \"$(printf '%s'
"${#a_ulock[@]}")\" system accounts that are not locked\n   - List of
accounts:\n$(printf '%s\n' "${a_ulock[@]:0:$l_limit}")\n   - end of list\n"
   fi
   unset a_users; unset a_ulock # Remove arrays
   if [ -z "$l_output2" ]; then
      echo -e "\n- Audit Result:\n  ** PASS **\n - * Correctly configured *
:\n$l_output\n"
   else
      echo -e "\n- Audit Result:\n  ** FAIL **\n - * Reasons for audit
failure * :\n$l_output2"
      [ -n "$l_output" ] && echo -e "- * Correctly configured *
:\n$l_output\n"
   fi
}
```

Note:

- The `root`, `sync`, `shutdown`, and `halt` users are exempted from requiring a non-login shell
- `root` is exempt from being locked

Remediation:

Set the shell for any accounts returned by the audit to nologin:

```
# usermod -s $(which nologin) <user>
```

Lock any non-root accounts returned by the audit:

```
# usermod -L <user>
```

The following script will:

- Set the shell for any accounts returned by the audit to nologin
- Lock any non-root system accounts returned by the audit

```
#!/usr/bin/env bash

{
   l_output="" l_output2=""
   l_valid_shells="^($( awk -F\/ '$NF != "nologin" {print}' /etc/shells | sed
-rn '/^\//{s,/,\\\\/,g;p}' | paste -s -d '|' - ))$"
   a_users=(); a_ulock=() # initialize arrays
   while read -r l_user; do # change system accounts that have a valid login
shell to nolog shell
      echo -e " - System account \"$l_user\" has a valid logon shell,
changing shell to \"$(which nologin)\""
      usermod -s "$(which nologin)" "$l_user"
   done < <(awk -v pat="$l_valid_shells" -F:
'($1!~/(root|sync|shutdown|halt|^\+)/ && $3<'"$(awk '/^\s*UID_MIN/{print $2}'
/etc/login.defs)"' && $(NF) ~ pat) { print $1 }' /etc/passwd)
   while read -r l_ulock; do # Lock system accounts that aren't locked
      echo -e " - System account \"$l_ulock\" is not locked, locking account"
      usermod -L "$l_ulock"
   done < <(awk -v pat="$l_valid_shells" -F: '($1!~/(root|^\+)/ && $2!~/LK?/
&& $3<'"$(awk '/^\s*UID_MIN/{print $2}' /etc/login.defs)"' && $(NF) ~ pat) {
print $1 }' /etc/passwd)
}
```

References:

1. NIST SP 800-53 Rev. 5: AC-2. AC-3, AC-5, MP-2

Additional Information:

The `root`, `sync`, `shutdown`, and `halt` users are exempted from requiring a non-login shell.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.5.3 Ensure default group for the root account is GID 0 (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `usermod` command can be used to specify which group the `root` account belongs to. This affects permissions of files that are created by the `root` account.

Rationale:

Using GID 0 for the `root` account helps prevent `root` -owned files from accidentally becoming accessible to non-privileged users.

Audit:

Run the following command and verify the result is `0` :

```
# grep "^root:" /etc/passwd | cut -f4 -d:
0
```

Remediation:

Run the following command to set the `root` account default group to GID `0` :

```
# usermod -g 0 root
```

References:

1. NIST SP 800-53 Rev. 5: CM-1,CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.1 Establish and Maintain a Secure Configuration Process Establish and maintain a secure configuration process for enterprise assets (end-user devices, including portable and mobile, non-computing/IoT devices, and servers) and software (operating systems and applications). Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	5.1 Establish Secure Configurations Maintain documented, standard security configuration standards for all authorized operating systems and software.	●	●	●

5.5.4 Ensure default user umask is 027 or more restrictive (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The user file-creation mode mask (`umask`) is use to determine the file permission for newly created directories and files. In Linux, the default permissions for any newly created directory is 0777 (`rwxrwxrwx`), and for any newly created file it is 0666 (`rw-rw-rw-`). The `umask` modifies the default Linux permissions by restricting (masking) these permissions. The `umask` is not simply subtracted, but is processed bitwise. Bits set in the `umask` are cleared in the resulting file mode.

`umask` can be set with either `octal` or `Symbolic` values:

- `Octal` (Numeric) Value - Represented by either three or four digits. ie `umask 0027` or `umask 027`. If a four digit umask is used, the first digit is ignored. The remaining three digits effect the resulting permissions for user, group, and world/other respectively.
- `Symbolic` Value - Represented by a comma separated list for User `u`, group `g`, and world/other `o`. The permissions listed are not masked by `umask`. ie a `umask` set by `umask u=rwx,g=rx,o=` is the `Symbolic` equivalent of the `Octal umask 027`. This `umask` would set a newly created directory with file mode `drwxr-x---` and a newly created file with file mode `rw-r-----`.

The default `umask` can be set to use the `pam_umask` module or in a `System Wide Shell Configuration File`. The user creating the directories or files has the discretion of changing the permissions via the chmod command, or choosing a different default `umask` by adding the `umask` command into a `User Shell Configuration File`, (`.bash_profile` or `.bashrc`), in their home directory.

Setting the default umask:

- pam_umask module:
  - will set the umask according to the system default in `/etc/login.defs` and user settings, solving the problem of different `umask` settings with different shells, display managers, remote sessions etc.
  - `umask=<mask>` value in the `/etc/login.defs` file is interpreted as Octal
  - Setting `USERGROUPS_ENAB` to yes in `/etc/login.defs` (default):
    - will enable setting of the `umask` group bits to be the same as owner bits. (examples: 022 -> 002, 077 -> 007) for non-root users, if the `uid` is the same as `gid`, and `username` is the same as the `<primary group name>`
    - userdel will remove the user's group if it contains no more members, and useradd will create by default a group with the name of the user
- `System Wide Shell Configuration File`:
  - `/etc/profile` - used to set system wide environmental variables on users shells. The variables are sometimes the same ones that are in the `.bash_profile`, however this file is used to set an initial PATH or PS1 for all shell users of the system. **is only executed for interactive *login* shells, or shells executed with the --login parameter.**
  - `/etc/profile.d` - `/etc/profile` will execute the scripts within `/etc/profile.d/*.sh`. It is recommended to place your configuration in a shell script within `/etc/profile.d` to set your own system wide environmental variables.
  - `/etc/bashrc` - System wide version of `.bashrc`. In Fedora derived distributions, `etc/bashrc` also invokes /etc/profile.d/*.sh if *non-login* shell, but redirects output to `/dev/null` if *non-interactive.* **Is only executed for *interactive* shells or if `BASH_ENV` is set to `/etc/bashrc`.**

User Shell Configuration Files:

- `~/.bash_profile` - Is executed to configure your shell before the initial command prompt. **Is only read by login shells.**
- `~/.bashrc` - Is executed for interactive shells. **only read by a shell that's both interactive and non-login**

Rationale:

Setting a secure default value for `umask` ensures that users make a conscious choice about their file permissions. A permissive `umask` value could result in directories or files with excessive permissions that can be read and/or written to by unauthorized users.

Audit:

Run the following to verify:

- A default user `umask` is set to enforce a newly created directories' permissions to be `750 (drwxr-x---)`, and a newly created file's permissions be `640 (rw-r-----)`, or more restrictive
- No less restrictive System Wide umask is set

Run the following script to verify that a default user umask is set enforcing a newly created directories's permissions to be 750 (drwxr-x---), and a newly created file's permissions be 640 (rw-r-----), or more restrictive:

```
#!/bin/bash

passing=""
grep -Eiq '^\s*UMASK\s+(0[0-7][2-7]7|[0-7][2-7]7)\b' /etc/login.defs && grep
-Eqi '^\s*USERGROUPS_ENAB\s*"?no"?\b' /etc/login.defs && grep -Eq
'^\s*session\s+(optional|requisite|required)\s+pam_umask\.so\b'
/etc/pam.d/common-session && passing=true
grep -REiq '^\s*UMASK\s+\s*(0[0-7][2-7]7|[0-7][2-
7]7|u=(r?|w?|x?)(r?|w?|x?)(r?|w?|x?),g=(r?x?|x?r?),o=)\b' /etc/profile*
/etc/bashrc* && passing=true
[ "$passing" = true ] && echo "Default user umask is set"
```

Verify output is: "Default user umask is set"
Run the following to verify that no less restrictive system wide umask is set:

```
# grep -RPi '(^|^[^#]*)\s*umask\s+([0-7][0-7][01][0-7]\b|[0-7][0-7][0-7][0-
6]\b|[0-7][01][0-7]\b|[0-7][0-7][0-
6]\b|(u=[rwx]{0,3},)?(g=[rwx]{0,3},)?o=[rwx]+\b|(u=[rwx]{1,3},)?g=[^rx]{1,3}(
,o=[rwx]{0,3})?\b)' /etc/login.defs /etc/profile* /etc/bashrc*

No file should be returned
```

Remediation:

Review /etc/bashrc, /etc/profile, and all files ending in *.sh in the /etc/profile.d/ directory and remove or edit all `umask` entries to follow local site policy. Any remaining entries should be: `umask 027`, `umask u=rwx,g=rx,o=` or more restrictive.
Configure `umask` in **one** of the following files:

- A file in the `/etc/profile.d/` directory ending in `.sh`
- `/etc/profile`
- `/etc/bashrc`

Example:

```
# vi /etc/profile.d/set_umask.sh

umask 027
```

Run the following command and remove or modify the `umask` of any returned files:

```
# grep -RPi '(^|^[^#]*)\s*umask\s+([0-7][0-7][01][0-7]\b|[0-7][0-7][0-7][0-
6]\b|[0-7][01][0-7]\b|[0-7][0-7][0-
6]\b|(u=[rwx]{0,3},)?(g=[rwx]{0,3},)?o=[rwx]+\b|(u=[rwx]{1,3},)?g=[^rx]{1,3}(
,o=[rwx]{0,3})?\b)' /etc/login.defs /etc/profile* /etc/bashrc*
```

Follow one of the following methods to set the default user umask:
Edit `/etc/login.defs` and edit the `UMASK` and `USERGROUPS_ENAB` lines as follows:

```
UMASK 027

USERGROUPS_ENAB no
```

Edit the files `/etc/pam.d/password-auth` and `/etc/pam.d/system-auth` and add or edit the following:

```
session     optional      pam_umask.so
```

OR Configure umask in one of the following files:

- A file in the `/etc/profile.d/` directory ending in .sh
- `/etc/profile`
- `/etc/bashrc`
  Example: /etc/profile.d/set_umask.sh

```
umask 027
```

Note: this method only applies to bash and shell. If other shells are supported on the system, it is recommended that their configuration files also are checked.

Default Value:

UMASK 022

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

Additional Information:

- Other methods of setting a default user umask exist
- If other methods are in use in your environment they should be audited
- The default user umask can be overridden with a user specific umask
- The user creating the directories or files has the discretion of changing the permissions:
    - Using the chmod command
    - Setting a different default umask by adding the umask command into a User Shell Configuration File, (.bashrc), in their home directory
    - Manually changing the umask for the duration of a login session by running the umask command

NIST SP 800-53 Rev. 5:

- AC-3
- MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6 Logging and Auditing

The items in this section describe how to configure logging, log monitoring, and auditing, using tools included in most distributions.

It is recommended that `rsyslog` be used for logging (with `logwatch` providing summarization) and `auditd` be used for auditing (with `aureport` providing summarization) to automatically monitor logs for intrusion attempts and other suspicious system behavior.

In addition to the local log files created by the steps in this section, it is also recommended that sites collect copies of their system logs on a secure, centralized log server via an encrypted connection. Not only does centralized logging help sites correlate events that may be occurring on multiple systems, but having a second copy of the system log information may be critical after a system compromise where the attacker has modified the local log files on the affected system(s). If a log correlation system is deployed, configure it to process the logs described in this section.

Because it is often necessary to correlate log information from many different systems (particularly after a security incident) it is recommended that the time be synchronized among systems and devices connected to the local network. The standard Internet protocol for time synchronization is the Network Time Protocol (NTP), which is supported by most network-ready devices. Reference <http://chrony.tuxfamily.org/> manual page for more information on configuring chrony.

It is important that all logs described in this section be monitored on a regular basis and correlated to determine trends. A seemingly innocuous entry in one log could be more significant when compared to an entry in another log.

Note on log file permissions: There really isn't a "one size fits all" solution to the permissions on log files. Many sites utilize group permissions so that administrators who are in a defined security group, such as "wheel" do not have to elevate privileges to root in order to read log files. Also, if a third-party log aggregation tool is used, it may need to have group permissions to read the log files, which is preferable to having it run setuid to root. Therefore, there are two remediation and audit steps for log file permissions. One is for systems that do not have a secured group method implemented that only permits root to read the log files (`root:root 600`). The other is for sites that do have such a setup and are designated as `root:securegrp 640` where `securegrp` is the defined security group (in some cases `wheel`).

6.1 System Logging

Logging services should be configured to prevent information leaks and to aggregate logs on a remote server so that they can be reviewed in the event of a system compromise. A centralized log server provides a single point of entry for further analysis, monitoring and filtering.

Security principals for logging

- Ensure transport layer security is implemented between the client and the log server.
- Ensure that logs are rotated as per the environment requirements.
- Ensure all locally generated logs have the appropriate permissions.
- Ensure all security logs are sent to a remote log server.
- Ensure the required events are logged.

What is covered

This section will cover the minimum best practices for the usage of ***either*** `rsyslog` ***or*** `journald`. The recommendations are written such that each is wholly independent of each other and ***only one is implemented***.

- If your organization makes use of an enterprise wide logging system completely outside of `rsyslog` or `journald`, then the following recommendations do not directly apply. However, the principals of the recommendations should be followed regardless of what solution is implemented. If the enterprise solution incorporates either of these tools, careful consideration should be given to the following recommendations to determine exactly what applies.
- Should your organization make use of both `rsyslog` and `journald`, take care how the recommendations may or may not apply to you.

What is not covered

- Enterprise logging systems not utilizing `rsyslog` or `journald`. As logging is very situational and dependent on the local environment, not everything can be covered here.
- Transport layer security should be applied to all remote logging functionality. Both `rsyslog` and `journald` supports secure transport and should be configured as such.
- The log server. There are a multitude of reasons for a centralized log server (and keeping a short period of logging on the local system), but the log server is out of scope for these recommendations.

6.1.1 Configure journald

Included in the systemd suite is a journaling service called `systemd-journald.service` for the collection and storage of logging data. It creates and maintains structured, indexed journals based on logging information that is received from a variety of sources such as:

- Classic RFC3164 BSD syslog via the /dev/log socket
- STDOUT/STDERR of programs via StandardOutput=journal + StandardError=journal in service files (both of which are default settings)
- Kernel log messages via the /dev/kmsg device node
- Audit records via the kernel's audit subsystem
- Structured log messages via journald's native protocol

Any changes made to the `systemd-journald` configuration will require a re-start of `systemd-journald`

6.1.1.1 Configure systemd-journald service

`systemd-journald` is a system service that collects and stores logging data. It creates and maintains structured, indexed journals based on logging information that is received from a variety of sources:

- Kernel log messages, via kmsg
- Simple system log messages, via the libc syslog call
- Structured system log messages via the native Journal API
- Standard output and standard error of service units
- Audit records, originating from the kernel audit subsystem

The daemon will implicitly collect numerous metadata fields for each log messages in a secure and unfakeable way. See systemd.journal-fields man page for more information about the collected metadata.

The journal service stores log data either persistently below `/var/log/journal` or in a volatile way below `/run/log/journal/`. By default, log data is stored persistently if `/var/log/journal/` exists during boot, with an implicit fallback to volatile storage. Use `Storage=` in `journald.conf` to configure where log data is placed, independently of the existence of `/var/log/journal/`.

On systems where `/var/log/journal/` does not exist but where persistent logging is desired, and the default `journald.conf` is used, it is sufficient to create the directory and ensure it has the correct access modes and ownership.

6.1.1.1.1 Ensure journald service is active (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Ensure that the `systemd-journald` service is enabled to allow capturing of logging events.

Rationale:

If the `systemd-journald` service is not enabled to start on boot, the system will not capture logging events.

Audit:

Run the following command to verify `systemd-journald` is enabled:

```
# systemctl is-enabled systemd-journald.service

static
```

Note: By default the `systemd-journald` service does not have an `[Install]` section and thus cannot be enabled / disabled. It is meant to be referenced as `Requires` or `Wants` by other unit files. As such, if the status of `systemd-journald` is not `static`, investigate why
Run the following command to verify `systemd-journald` is active:

```
# systemctl is-active systemd-journald.service

active
```

Remediation:

Run the following commands to unmask and start `systemd-journald.service`

```
# systemctl unmask systemd-journald.service
# systemctl start systemd-journald.service
```

References:

1. NIST SP 800-53 :: SC-24
2. NIST SP 800-53A :: SC-24.1 (v)
3. STIG ID: RHEL-09-211040 | RULE ID: SV-257783r991562 | CAT II

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.1.1.1.2 Ensure journald log file access is configured (Manual)

Profile Applicability:

- Level 1 - Server

Description:

Journald will create logfiles that do not already exist on the system. This setting controls what permissions will be applied to these newly created files.

Rationale:

It is important to ensure that log files have the correct permissions to ensure that sensitive data is archived and protected.

Audit:

First determine if there is an override file `/etc/tmpfiles.d/systemd.conf`. If so, this file will override all default settings as defined in `/usr/lib/tmpfiles.d/systemd.conf` and should be inspected.
If no override file exists, inspect the default `/usr/lib/tmpfiles.d/systemd.conf` against the site specific requirements.
Ensure that file permissions are mode `0640` or more restrictive.
Run the following script to verify if an override file exists or not and if the files permissions are mode `640` or more restrictive:

```
#!/usr/bin/env bash

{
    l_output="" file_path=""
    # Check for the existence of an override file
    if [ -f /etc/tmpfiles.d/systemd.conf ]; then
        file_path="/etc/tmpfiles.d/systemd.conf"
    elif [ -f /usr/lib/tmpfiles.d/systemd.conf ]; then
        file_path="/usr/lib/tmpfiles.d/systemd.conf"
    fi
    if [ -n "$file_path" ]; then # Ensure a file path is found
        higher_permissions_found=false  # Initialize a flag to check if
higher permissions are found
        # Read the file line by line and check for permissions higher than
0640
        while IFS= read -r line; do
            if echo "$line" | grep -Piq '^\s*[a-z]+\s+[^\s]+\s+0*([6-7][4-
7][1-7]|7[0-7][0-7])\s+'; then
                higher_permissions_found=true
                break
            fi
        done < "$file_path"
        if $higher_permissions_found; then
            echo -e "\n - permissions other than 0640 found in $file_path"
                        l_output="$l_output\n - Inspect $file_path"
        else
            echo -e "All permissions inside $file_path are 0640 or more
restrictive."
        fi
    fi
   if [ -z "$l_output" ]; then # Provide output from checks
      echo -e "\n- Audit Result:\n  ** PASS **\n$file_path exists and has
correct permissions set\n"
   else
      echo -e "\n- Audit Result:\n  ** REVIEW **\n$l_output\n - Review
permissions to ensure they are set IAW site policy"
   fi
}
```

Remediation:

If the default configuration is not appropriate for the site specific requirements, copy `/usr/lib/tmpfiles.d/systemd.conf` to `/etc/tmpfiles.d/systemd.conf` and modify as required. Requirements is either `0640` or site policy if that is less restrictive.

References:

1. NIST SP 800-53 Rev. 5: AC-3, AU-2, AU-12, MP-2, SI-5

Additional Information:

See `man 5 tmpfiles.d` for detailed information on the permission sets for the relevant log files. Further information with examples can be found at https://www.freedesktop.org/software/systemd/man/tmpfiles.d.html

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.1.1.1.3 Ensure journald ForwardToSyslog is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Data from `journald` should be kept in the confines of the service and not forwarded to other services.

Rationale:

- IF - `journald` is the method for capturing logs, all logs of the system should be handled by `journald` and not forwarded to other logging mechanisms.

Note: This recommendation **only applies if `journald` is the chosen method for client side logging**. Do not apply this recommendation if `rsyslog` is used.

Audit:

- IF - `journald` is the method for capturing logs
Run the following script to verify `ForwardToSyslog` in not set to `yes`:

```
#!/usr/bin/env bash

{
   if systemctl is-active rsyslog.service | grep -Psq -- '^active'; then
      l_setting="yes"
   else
      l_setting="no"
   fi
   l_parameters="systemd/journald.conf:Journal:ForwardToSyslog:$l_setting"
   l_analyze_cmd="$(readlink -e /bin/systemd-analyze || readlink -e
/usr/bin/systemd-analyze)"
   a_output=() a_output2=() a_output3=() l_out="" l_out2="" l_opt=""

   f_pass_output()
   {
      if [ "${#a_output[@]}" -gt 0 ] || [ "${#a_output2[@]}" -gt 0 ]; then
         a_output3+=("  - $l_option is correctly set to $l_option_value" "
in $l_file" \
         "    but this setting will be ignored do to load preference")
      else
         if [ -n "$l_out2" ]; then
            a_output+=("$l_out2" "  - Default for $l_option is correctly set
to $l_option_value" "$l_out")
         else
            a_output+=("  - $l_option is correctly set to $l_option_value" "
in $l_file" "$l_out")
         fi
      fi
   }

   f_fail_output()
   {
      if [ "${#a_output[@]}" -gt 0 ] || [ "${#a_output2[@]}" -gt 0 ]; then
         a_output3+=("  - $l_option is incorrectly set to $l_option_value" "
in $l_file" \
         "    but this setting will be ignored do to load preference")
      else
         if [ -n "$l_out2" ]; then
            a_output2+=("$l_out2" "  - Default for $l_option is incorrectly
set to $l_option_value" "$l_out")
         else
            a_output2+=("  - $l_option is incorrectly set to $l_option_value"
"    in $l_file" "$l_out")
         fi
      fi
   }

   f_option_chk()
   {

            if [ "$l_option_value" = "$l_value" ]; then
               f_pass_output
            else
               f_fail_output
            fi
   }
```

```
   while IFS=: read -r l_conf_file l_block l_option l_value; do
      l_out="    and should be equal to $l_value"
      while IFS= read -r l_file; do
         l_file="${l_file//# /}"
         l_opt="$(awk '/\['"$l_block"'\]/{a=1;next}/\[/{a=0}a' "$l_file"
2>/dev/null | grep -Poi '^\h*'"$l_option"'\h*=\h*\H+\b' | tail -n 1)"
         if [ -n "$l_opt" ]; then
            l_option_value="$(cut -d= -f2 <<< "$l_opt" | xargs)"
            f_option_chk
         fi
      done < <("$l_analyze_cmd" cat-config "$l_conf_file" | tac | grep -Pio
'^\h*#\h*\/[^#\n\r\h]+\.conf\b')
      # If nothing is explicitly set, check default
      if [ "${#a_output[@]}" -le 0 ] && [ "${#a_output2[@]}" -le 0 ]; then
         l_file="/etc/$l_conf_file"
         l_opt="$(awk '/\['"$l_block"'\]/{a=1;next}/\[/{a=0}a' "$l_file"
2>/dev/null | grep -Poim 1 '^(\h*#)?\h*'"$l_option"'\h*=\h*\H+\b')"
         if [ -n "$l_opt" ]; then
            l_option_value="$(cut -d= -f2 <<< "${l_opt//# /}" | xargs)"
            l_out2=" - Note: default value \"${l_opt//#/}\" is being used in
the configuration"
            f_option_chk
         fi
      fi
   done <<< "$l_parameters"

   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
      [ "${#a_output3[@]}" -gt 0 ] && printf '%s\n' "  ** Note: **"
"${a_output3[@]}" ""
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "" "- Correctly set:"
"${a_output[@]}" ""
      [ "${#a_output3[@]}" -gt 0 ] && printf '%s\n' "  ** Note: **"
"${a_output3[@]}" ""
   fi
}
```

Remediation:

- IF - `journald` is the preferred method for capturing logs:
Set the following parameter in the `[Journal]` section in `/etc/systemd/journald.conf` or a file in /etc/systemd/journald.conf.d/ ending in `.conf`:

```
ForwardToSyslog=no
```

Example:

```
#!/usr/bin/env bash

{
   [ ! -d /etc/systemd/journald.conf.d/ ] && mkdir
/etc/systemd/journald.conf.d/
   if grep -Psq -- '^\h*\[Journal\] /etc/systemd/journald.conf.d/60-
journald.conf; then
      printf '%s\n' "ForwardToSyslog=no" >> /etc/systemd/journald.conf.d/60-
journald.conf
   else
      printf '%s\n' "[Journal]" "ForwardToSyslog=no" >>
/etc/systemd/journald.conf.d/60-journald.conf
   fi
}
```

- ELSEIF - `rsyslog` is the preferred method for capturing logs:
Set the following parameter in the `[Journal]` section in `/etc/systemd/journald.conf` or a file in /etc/systemd/journald.conf.d/ ending in `.conf`:

```
ForwardToSyslog=yes
```

Example:

```
#!/usr/bin/env bash

{
   [ ! -d /etc/systemd/journald.conf.d/ ] && mkdir
/etc/systemd/journald.conf.d/
   if grep -Psq -- '^\h*\[Journal\] /etc/systemd/journald.conf.d/60-
journald.conf; then
      printf '%s\n' "ForwardToSyslog=yes" >> /etc/systemd/journald.conf.d/60-
journald.conf
   else
      printf '%s\n' "[Journal]" "ForwardToSyslog=yes" >>
/etc/systemd/journald.conf.d/60-journald.conf
   fi
}
```

Note: If this setting appears in a canonically later file, or later in the same file, the setting will be overwritten
Run to following command to update the parameters in the service:

```
# systemctl reload-or-restart systemd-journald
```

Default Value:

ForwardToSyslog=no

References:

1. NIST SP 800-53 Rev. 5: AU-2, AU-6, AU-7, AU-12

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.1.1.1.4 Ensure systemd-journal-remote service is not in use (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Journald `systemd-journal-remote` supports the ability to receive messages from remote hosts, thus acting as a log server. Clients should not receive data from other hosts.

NOTE:

- The same package, `systemd-journal-remote`, is used for both sending logs to remote hosts and receiving incoming logs.
- With regards to receiving logs, there are two services; `systemd-journal-remote.socket` and `systemd-journal-remote.service`.

Rationale:

If a client is configured to also receive data, thus turning it into a server, the client system is acting outside it's operational boundary.

Audit:

Run the following command to verify `systemd-journal-remote.socket` and `systemd-journal-remote.service` are not enabled:

```
# systemctl is-enabled systemd-journal-remote.socket systemd-journal-
remote.service | grep -P -- '^enabled'

Nothing should be returned
```

Run the following command to verify `systemd-journal-remote.socket` and `systemd-journal-remote.service` are not active:

```
# systemctl is-active systemd-journal-remote.socket systemd-journal-
remote.service | grep -P -- '^active'

Nothing should be returned
```

Remediation:

Run the following commands to stop and mask `systemd-journal-remote.socket` and systemd-journal-remote.service:

```
# systemctl stop systemd-journal-remote.socket systemd-journal-remote.service
# systemctl mask systemd-journal-remote.socket systemd-journal-remote.service
```

References:

1. NIST SP 800-53 Rev. 5: AU-2, AU-7 AU-12

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

6.1.1.1.5 Ensure journald Storage is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Data from journald may be stored in volatile memory or persisted locally on the server. Logs in memory will be lost upon a system reboot. By persisting logs to local disk on the server they are protected from loss due to a reboot.

Rationale:

Writing log data to disk will provide the ability to forensically reconstruct events which may have impacted the operations or security of a system even after a system crash or reboot.

Audit:

Run the following script to verify `Storage` is set to `persistent`:

```
#!/usr/bin/env bash

{
   l_output="" l_output2=""
   a_parlist=("Storage=persistent")
   l_systemd_config_file="/etc/systemd/journald.conf" # Main systemd configuration
file
   config_file_parameter_chk()
   {
      unset A_out; declare -A A_out # Check config file(s) setting
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_systemd_parameter="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               grep -Piq -- "^\h*$l_systemd_parameter_name\b" <<<
"$l_systemd_parameter" && A_out+=(["$l_systemd_parameter"]="$l_file")
            fi
         fi
      done < <(/usr/bin/systemd-analyze cat-config "$l_systemd_config_file" | grep -
Pio '^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate output
         while IFS="=" read -r l_systemd_file_parameter_name
l_systemd_file_parameter_value; do
            l_systemd_file_parameter_name="${l_systemd_file_parameter_name// /}"
            l_systemd_file_parameter_value="${l_systemd_file_parameter_value// /}"
            if grep -Piq "^\h*$l_systemd_parameter_value\b" <<<
"$l_systemd_file_parameter_value"; then
               l_output="$l_output\n - \"$l_systemd_parameter_name\" is correctly set
to \"$l_systemd_file_parameter_value\" in \"$(printf '%s' "${A_out[@]}")\"\n"
            else
               l_output2="$l_output2\n - \"$l_systemd_parameter_name\" is incorrectly
set to \"$l_systemd_file_parameter_value\" in \"$(printf '%s' "${A_out[@]}")\" and
should have a value matching: \"$l_systemd_parameter_value\"\n"
            fi
         done < <(grep -Pio -- "^\h*$l_systemd_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         l_output2="$l_output2\n - \"$l_systemd_parameter_name\" is not set in an
included file\n   ** Note: \"$l_systemd_parameter_name\" May be set in a file that's
ignored by load procedure **\n"
      fi
   }
   while IFS="=" read -r l_systemd_parameter_name l_systemd_parameter_value; do #
Assess and check parameters
      l_systemd_parameter_name="${l_systemd_parameter_name// /}"
      l_systemd_parameter_value="${l_systemd_parameter_value// /}"
      config_file_parameter_chk
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ -z "$l_output2" ]; then # Provide output from checks
      echo -e "\n- Audit Result:\n  ** PASS **\n$l_output\n"
   else
      echo -e "\n- Audit Result:\n  ** FAIL **\n - Reason(s) for audit
failure:\n$l_output2"
      [ -n "$l_output" ] && echo -e "\n- Correctly set:\n$l_output\n"
   fi
}
```

Remediation:

Set the following parameter in the `[Journal]` section in `/etc/systemd/journald.conf` or a file in `/etc/systemd/journald.conf.d/` ending in `.conf`:

```
Storage=persistent
```

Example:

```
#!/usr/bin/env bash

{
   [ ! -d /etc/systemd/journald.conf.d/ ] && mkdir
/etc/systemd/journald.conf.d/
   if grep -Psq -- '^\h*\[Journal\]' /etc/systemd/journald.conf.d/60-
journald.conf; then
      printf '%s\n' "Storage=persistent" >> /etc/systemd/journald.conf.d/60-
journald.conf
   else
      printf '%s\n' "[Journal]" "Storage=persistent" >>
/etc/systemd/journald.conf.d/60-journald.conf
   fi
}
```

Note: If this setting appears in a canonically later file, or later in the same file, the setting will be overwritten
Run to following command to update the parameters in the service:

```
# systemctl reload-or-restart systemd-journald
```

References:

1. NIST SP 800-53 Rev. 5: AU-3, AU-12

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.1.1.1.6 Ensure journald Compress is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The journald system includes the capability of compressing overly large files to avoid filling up the system with logs or making the logs unmanageably large.

Rationale:

Uncompressed large files may unexpectedly fill a filesystem leading to resource unavailability. Compressing logs prior to write can prevent sudden, unexpected filesystem impacts.

Audit:

Run the following script to verify `Compress` is set to `yes`:

```
#!/usr/bin/env bash

{
   l_output="" l_output2=""
   a_parlist=("Compress=yes")
   l_systemd_config_file="/etc/systemd/journald.conf" # Main systemd configuration
file
   config_file_parameter_chk()
   {
      unset A_out; declare -A A_out # Check config file(s) setting
      while read -r l_out; do
         if [ -n "$l_out" ]; then
            if [[ $l_out =~ ^\s*# ]]; then
               l_file="${l_out//# /}"
            else
               l_systemd_parameter="$(awk -F= '{print $1}' <<< "$l_out" | xargs)"
               grep -Piq -- "^\h*$l_systemd_parameter_name\b" <<<
"$l_systemd_parameter" && A_out+=(["$l_systemd_parameter"]="$l_file")
            fi
         fi
      done < <(/usr/bin/systemd-analyze cat-config "$l_systemd_config_file" | grep -
Pio '^\h*([^#\n\r]+|#\h*\/[^#\n\r\h]+\.conf\b)')
      if (( ${#A_out[@]} > 0 )); then # Assess output from files and generate output
         while IFS="=" read -r l_systemd_file_parameter_name
l_systemd_file_parameter_value; do
            l_systemd_file_parameter_name="${l_systemd_file_parameter_name// /}"
            l_systemd_file_parameter_value="${l_systemd_file_parameter_value// /}"
            if grep -Piq "^\h*$l_systemd_parameter_value\b" <<<
"$l_systemd_file_parameter_value"; then
               l_output="$l_output\n - \"$l_systemd_parameter_name\" is correctly set
to \"$l_systemd_file_parameter_value\" in \"$(printf '%s' "${A_out[@]}")\"\n"
            else
               l_output2="$l_output2\n - \"$l_systemd_parameter_name\" is incorrectly
set to \"$l_systemd_file_parameter_value\" in \"$(printf '%s' "${A_out[@]}")\" and
should have a value matching: \"$l_systemd_parameter_value\"\n"
            fi
         done < <(grep -Pio -- "^\h*$l_systemd_parameter_name\h*=\h*\H+"
"${A_out[@]}")
      else
         l_output2="$l_output2\n - \"$l_systemd_parameter_name\" is not set in an
included file\n   ** Note: \"$l_systemd_parameter_name\" May be set in a file that's
ignored by load procedure **\n"
      fi
   }
   while IFS="=" read -r l_systemd_parameter_name l_systemd_parameter_value; do #
Assess and check parameters
      l_systemd_parameter_name="${l_systemd_parameter_name// /}"
      l_systemd_parameter_value="${l_systemd_parameter_value// /}"
      config_file_parameter_chk
   done < <(printf '%s\n' "${a_parlist[@]}")
   if [ -z "$l_output2" ]; then # Provide output from checks
      echo -e "\n- Audit Result:\n  ** PASS **\n$l_output\n"
   else
      echo -e "\n- Audit Result:\n  ** FAIL **\n - Reason(s) for audit
failure:\n$l_output2"
      [ -n "$l_output" ] && echo -e "\n- Correctly set:\n$l_output\n"
   fi
}
```

Remediation:

Set the following parameter in the `[Journal]` section in `/etc/systemd/journald.conf` or a file in `/etc/systemd/journald.conf.d/` ending in `.conf`:

```
Compress=yes
```

Example:

```
#!/usr/bin/env bash

{
   [ ! -d /etc/systemd/journald.conf.d/ ] && mkdir
/etc/systemd/journald.conf.d/
   if grep -Psq -- '^\h*\[Journal\]' /etc/systemd/journald.conf.d/60-
journald.conf; then
      printf '%s\n' "Compress=yes" >> /etc/systemd/journald.conf.d/60-
journald.conf
   else
      printf '%s\n' "[Journal]" "Compress=yes" >>
/etc/systemd/journald.conf.d/60-journald.conf
   fi
}
```

Note: If this setting appears in a canonically later file, or later in the same file, the setting will be overwritten
Run to following command to update the parameters in the service:

```
# systemctl reload-or-restart systemd-journald
```

References:

1. NIST SP 800-53 Rev. 5: AU-4

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v8	8.3 Ensure Adequate Audit Log Storage Ensure that logging destinations maintain adequate storage to comply with the enterprise's audit log management process.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●
v7	6.4 Ensure adequate storage for logs Ensure that all systems that store logs have adequate storage space for the logs generated.		●	●

6.1.2 Configure rsyslog

The `rsyslog` software package may be used instead of the default `journald` logging mechanism.

Rsyslog has evolved over several decades. For this reason it supports three different configuration formats ("languages"):

- `basic` - previously known as the `sysklogd` format, this is the format best used to express basic things, such as where the statement fits on a single line.
    - It stems back to the original syslog.conf format, in use now for several decades.
    - The most common use case is matching on facility/severity and writing matching messages to a log file.
- `advanced` - previously known as the `RainerScript` format, this format was first available in rsyslog v6 and is the current, best and most precise format for non-trivial use cases where more than one line is needed.
    - Prior to v7, there was a performance impact when using this format that encouraged use of the basic format for best results. Current versions of rsyslog do not suffer from this (historical) performance impact.
    - This new style format is specifically targeted towards more advanced use cases like forwarding to remote hosts that might be partially offline.
- `obsolete legacy` - previously known simply as the `legacy` format, this format is exactly what its name implies: it is obsolete and should not be used when writing new configurations. It was created in the early days (up to rsyslog version 5) where we expected that rsyslog would extend sysklogd just mildly. Consequently, it was primarily aimed at small additions to the original sysklogd format.
    - Practice has shown that it was notoriously hard to use for more advanced use cases, and thus we replaced it with the advanced format.
    - In essence, everything that needs to be written on a single line that starts with a dollar sign is legacy format. Users of this format are encouraged to migrate to the basic or advanced formats.

Note: This section only applies if `rsyslog` is the chosen method for client side logging. Do not apply this section if `journald` is used.

6.1.2.1 Ensure rsyslog service is enabled and active (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Once the `rsyslog` package is installed, ensure that the service is enabled.

Rationale:

If the `rsyslog` service is not enabled to start on boot, the system will not capture logging events.

Audit:

- IF - `rsyslog` is being used for logging on the system:
Run the following command to verify `rsyslog.service` is enabled:

```
# systemctl is-enabled rsyslog

enabled
```

Run the following command to verify `rsyslog.service` is active:

```
# systemctl is-active rsyslog.service

active
```

Remediation:

- IF - `rsyslog` is being used for logging on the system:
Run the following commands to unmask, enable, and start `rsyslog.service`:

```
# systemctl unmask rsyslog.service
# systemctl enable rsyslog.service
# systemctl start rsyslog.service
```

References:

1. NIST SP 800-53 Rev. 5: AU-2, AU-3, AU-12

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.1.2.2 Ensure rsyslog log file creation mode is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`rsyslog` will create logfiles that do not already exist on the system.

The `global()` configuration object `umask`, available in rsyslog 8.26.0+, sets the rsyslogd process' umask. If not specified, the system-provided default is used. The value given must always be a 4-digit octal number, with the initial digit being zero.

The legacy `$umask` parameter sets the `rsyslogd` process' umask. If not specified, the system-provided default is used. The value given must always be a 4-digit octal number, with the initial digit being zero.

The legacy `$FileCreateMode` parameter allows the setting of the mode with which `rsyslogd` creates new files. If not specified, the value `0644` is used. The value given must always be a 4-digit octal number, with the initial digit being zero. Please note that the actual permission depend on `rsyslogd` process `umask`. If in doubt, use `$umask 0000` right at the beginning of the configuration file to remove any restrictions.

The legacy `$FileCreateMode` may be specified multiple times. If so, it specifies the creation mode for all selector lines that follow until the next `$FileCreateMode` parameter. Order of lines is vitally important.

Rationale:

It is important to ensure that log files have the correct permissions to ensure that sensitive data is archived and protected.

Audit:

Run the following command
Run the following command to verify `$FileCreateMode`:

```
# grep -Ps '^\h*\$FileCreateMode\h+0[0,2,4,6][0,2,4]0\b' /etc/rsyslog.conf
/etc/rsyslog.d/*.conf
```

Verify the output is includes 0640 or more restrictive:

```
$FileCreateMode 0640
```

Should a site policy dictate less restrictive permissions, ensure to follow said policy.
NOTE: More restrictive permissions such as `0600` is implicitly sufficient.

Remediation:

Edit either `/etc/rsyslog.conf` or a dedicated `.conf` file in `/etc/rsyslog.d/` and set `$FileCreateMode` to `0640` or more restrictive:

```
$FileCreateMode 0640
```

Restart the service:

```
# systemctl restart rsyslog
```

References:

1. RSYSLOG.CONF(5)
2. NIST SP 800-53 Rev. 5: AC-3, AC-6, MP-2
3. https://www.rsyslog.com/doc/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v8	8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets.	●	●	●
v7	5.1 Establish Secure Configurations Maintain documented, standard security configuration standards for all authorized operating systems and software.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.		●	●

6.1.2.3 Ensure rsyslog is not configured to receive logs from a remote client (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`rsyslog` supports the ability to receive messages from remote hosts, thus acting as a log server. Clients should not receive data from other hosts.

Rationale:

If a client is configured to also receive data, thus turning it into a server, the client system is acting outside its operational boundary.

Audit:

Review the `/etc/rsyslog.conf` and `/etc/rsyslog.d/*.conf` files and verify that the system is not configured to accept incoming logs.
advanced format

```
# grep -Psi -- '^\h*module\(load=\"?imtcp\"?\)' /etc/rsyslog.conf
/etc/rsyslog.d/*.conf
# grep -Psi -- '^\h*input\(type=\"?imtcp\"?\b' /etc/rsyslog.conf
/etc/rsyslog.d/*.conf
```

Nothing should be returned
obsolete legacy format

```
# grep -Psi -- '^\h*\$ModLoad\h+imtcp\b' /etc/rsyslog.conf
/etc/rsyslog.d/*.conf
# grep -Psi -- '^\h*\$InputTCPServerRun\b' /etc/rsyslog.conf
/etc/rsyslog.d/*.conf
```

Nothing should be returned

Remediation:

Should there be any active log server configuration found in the auditing section, modify those files and remove the specific lines highlighted by the audit. Verify none of the following entries are present in any of `/etc/rsyslog.conf` or `/etc/rsyslog.d/*.conf`.

advanced format

```
module(load="imtcp")
input(type="imtcp" port="514")
```

deprecated legacy format

```
$ModLoad imtcp
$InputTCPServerRun
```

Restart the service:

```
# systemctl restart rsyslog
```

References:

1. NIST SP 800-53 Rev. 5: AU-2, AU-7, AU-12, CM-6
2. https://www.rsyslog.com/doc/index.html

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

6.1.3 Configure Logfiles

6.1.3.1 Ensure access to all logfiles has been configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Log files contain information from many services on the the local system, or in the event of a centralized log server, others systems logs as well.

In general log files are found in `/var/log/`, although application can be configured to store logs elsewhere. Should your application store logs in another, ensure to run the same test on that location.

Rationale:

It is important that log files have the correct permissions to ensure that sensitive data is protected and that only the appropriate users / groups have access to them.

Audit:

Run the following script to verify that files in `/var/log/` have appropriate permissions and ownership:

```
#!/usr/bin/env bash

{
   a_output=(); a_output2=()
   f_file_test_chk()
   {
      a_out2=()
      maxperm="$( printf '%o' $(( 0777 & ~$perm_mask)) )"
      [ $(( $l_mode & $perm_mask )) -gt 0 ] && \
         a_out2+=("   o Mode: \"$l_mode\" should be \"$maxperm\" or more restrictive")
      [[ ! "$l_user" =~ $l_auser ]] && \
         a_out2+=("   o Owned by: \"$l_user\" and should be owned by \"${l_auser//|/ or }\"")
      [[ ! "$l_group" =~ $l_agroup ]] && \
         a_out2+=("   o Group owned by: \"$l_group\" and should be group owned by
\"${l_agroup//|/ or }\"")
      [ "${#a_out2[@]}" -gt 0 ] && a_output2+=(" - File: \"$l_fname\" is:" "${a_out2[@]}")
   }
   while IFS= read -r -d $'\0' l_file; do
      while IFS=: read -r l_fname l_mode l_user l_group; do
         if grep -Pq -- '\/(apt)\h*$' <<< "$(dirname "$l_fname")"; then
            perm_mask='0133' l_auser="root" l_agroup="(root|adm)"; f_file_test_chk
         else
            case "$(basename "$l_fname")" in
               lastlog | lastlog.* | wtmp | wtmp.* | wtmp-* | btmp | btmp.* | btmp-* | README)
                  perm_mask='0113' l_auser="root" l_agroup="(root|utmp)"
                  f_file_test_chk ;;
               cloud-init.log* | localmessages* | waagent.log*)
                  perm_mask='0133' l_auser="(root|syslog)" l_agroup="(root|adm)"
                  file_test_chk ;;
               secure{,*.*,.*,-*} | auth.log | syslog | messages)
                  perm_mask='0137' l_auser="(root|syslog)" l_agroup="(root|adm)"
                  f_file_test_chk ;;
               SSSD | sssd)
                  perm_mask='0117' l_auser="(root|SSSD)" l_agroup="(root|SSSD)"
                  f_file_test_chk ;;
               gdm | gdm3)
                  perm_mask='0117' l_auser="root" l_agroup="(root|gdm|gdm3)"
                  f_file_test_chk ;;
               *.journal | *.journal~)
                  perm_mask='0137' l_auser="root" l_agroup="(root|systemd-journal)"
                  f_file_test_chk ;;
               *)
                  perm_mask='0133' l_auser="(root|syslog)" l_agroup="(root|adm)"
                  if [ "$l_user" = "root" ] || ! grep -Pq -- "^\h*$(awk -F: '$1=="'"$l_user"'"
{print $7}' /etc/passwd)\b" /etc/shells; then
                     ! grep -Pq -- "$l_auser" <<< "$l_user" && l_auser="(root|syslog|$l_user)"
                     ! grep -Pq -- "$l_agroup" <<< "$l_group" && l_agroup="(root|adm|$l_group)"
                  fi
                  f_file_test_chk ;;
            esac
         fi
      done < <(stat -Lc '%n:%#a:%U:%G' "$l_file")
   done < <(find -L /var/log -type f \( -perm /0137 -o ! -user root -o ! -group root \) -print0)
   if [ "${#a_output2[@]}" -le 0 ]; then
      a_output+=("  - All files in \"/var/log/\" have appropriate permissions and ownership")
      printf '\n%s' "- Audit Result:" "  ** PASS **" "${a_output[@]}" ""
   else
      printf '\n%s' "- Audit Result:" "  ** FAIL **" " - Reason(s) for audit failure:"
"${a_output2[@]}" ""
   fi
}
```

Remediation:

Run the following script to update permissions and ownership on files in `/var/log`. Although the script is not destructive, ensure that the output is captured in the event that the remediation causes issues.

```
#!/usr/bin/env bash

{
   a_output2=()
   f_file_test_fix()
   {
      a_out2=()
      maxperm="$( printf '%o' $(( 0777 & ~$perm_mask)) )"
      if [ $(( $l_mode & $perm_mask )) -gt 0 ]; then
         a_out2+=("   o Mode: \"$l_mode\" should be \"$maxperm\" or more
restrictive" "     x Removing excess permissions")
         chmod "$l_rperms" "$l_fname"
      fi
      if [[ ! "$l_user" =~ $l_auser ]]; then
         a_out2+=("   o Owned by: \"$l_user\" and should be owned by
\"${l_auser//|/ or }\"" "     x Changing ownership to: \"$l_fix_account\"")
         chown "$l_fix_account" "$l_fname"
      fi
      if [[ ! "$l_group" =~ $l_agroup ]]; then
         a_out2+=("   o Group owned by: \"$l_group\" and should be group
owned by \"${l_agroup//|/ or }\"" "     x Changing group ownership to:
\"$l_fix_account\"")
         chgrp "$l_fix_account" "$l_fname"
      fi
      [ "${#a_out2[@]}" -gt 0 ] && a_output2+=(" - File: \"$l_fname\" is:"
"${a_out2[@]}")
   }
   l_fix_account='root'
   while IFS= read -r -d $'\0' l_file; do
      while IFS=: read -r l_fname l_mode l_user l_group; do
         if grep -Pq -- '\/(apt)\h*$' <<< "$(dirname "$l_fname")"; then
            perm_mask='0133' l_rperms="u-x,go-wx" l_auser="root"
l_agroup="(root|adm)"; f_file_test_fix
         else
            case "$(basename "$l_fname")" in
               lastlog | lastlog.* | wtmp | wtmp.* | wtmp-* | btmp | btmp.* |
btmp-* | README)
                  perm_mask='0113' l_rperms="ug-x,o-wx" l_auser="root"
l_agroup="(root|utmp)"
                  f_file_test_fix ;;
               cloud-init.log* | localmessages* | waagent.log*)
                  perm_mask='0133' l_rperms="u-x,go-wx"
l_auser="(root|syslog)" l_agroup="(root|adm)"
                  file_test_fix ;;
               secure | auth.log | syslog | messages)
                  perm_mask='0137' l_rperms="u-x,g-wx,o-rwx"
l_auser="(root|syslog)" l_agroup="(root|adm)"
                  f_file_test_fix ;;
               SSSD | sssd)
                  perm_mask='0117' l_rperms="ug-x,o-rwx"
l_auser="(root|SSSD)" l_agroup="(root|SSSD)"
                  f_file_test_fix ;;
               gdm | gdm3)
                  perm_mask='0117' l_rperms="ug-x,o-rwx" l_auser="root"
l_agroup="(root|gdm|gdm3)"
                  f_file_test_fix ;;
               *.journal | *.journal~)
```

```
                  perm_mask='0137' l_rperms="u-x,g-wx,o-rwx" l_auser="root"
l_agroup="(root|systemd-journal)"
                  f_file_test_fix ;;
               *)
                  perm_mask='0133' l_rperms="u-x,g-wx,o-rwx"
l_auser="(root|syslog)" l_agroup="(root|adm)"
                  if [ "$l_user" = "root" ] || ! grep -Pq -- "^\h*$(awk -F:
'$1=="'"$l_user"'" {print $7}' /etc/passwd)\b" /etc/shells; then
                     ! grep -Pq -- "$l_auser" <<< "$l_user" &&
l_auser="(root|syslog|$l_user)"
                     ! grep -Pq -- "$l_agroup" <<< "$l_group" &&
l_agroup="(root|adm|$l_group)"
                  fi
                  f_file_test_fix ;;
            esac
         fi
      done < <(stat -Lc '%n:%#a:%U:%G' "$l_file")
   done < <(find -L /var/log -type f \( -perm /0137 -o ! -user root -o ! -
group root \) -print0)
   if [ "${#a_output2[@]}" -le 0 ]; then # If all files passed, then we
report no changes
      a_output+=("  - All files in \"/var/log/\" have appropriate permissions
and ownership")
      printf '\n%s' "- All files in \"/var/log/\" have appropriate
permissions and ownership" "  o No changes required" ""
   else
      printf '\n%s' "${a_output2[@]}" ""
   fi
}
```

Note: You may also need to change the configuration for your logging software or services for any logs that had incorrect permissions.
If there are services that log to other locations, ensure that those log files have the appropriate access configured.

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.2 Ensure logrotate is configured (Manual)

Profile Applicability:

- Level 1 - Server

Description:

The system includes the capability of rotating log files regularly to avoid filling up the system with logs or making the logs unmanageably large. The file `/etc/logrotate.d/syslog` is the configuration file used to rotate log files created by `syslog` or `rsyslog`.

Rationale:

By keeping the log files smaller and more manageable, a system administrator can easily archive these files to another system and spend less time looking through inordinately large log files.

Audit:

Review `/etc/logrotate.conf` and `/etc/logrotate.d/*` and verify logs are rotated according to site policy.

Remediation:

Edit `/etc/logrotate.conf` and `/etc/logrotate.d/*` to ensure logs are rotated according to site policy.

References:

1. NIST SP 800-53 Rev. 5: AU-8

Additional Information:

If no `maxage` setting is set for `logrotate` a situation can occur where `logrotate` is interrupted and fails to delete rotated log files. It is recommended to set this to a value greater than the longest any log file should exist on your system to ensure that any such log file is removed but standard rotation settings are not overridden.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.3 Ensure Adequate Audit Log Storage Ensure that logging destinations maintain adequate storage to comply with the enterprise's audit log management process.	●	●	●
v7	6.4 Ensure adequate storage for logs Ensure that all systems that store logs have adequate storage space for the logs generated.		●	●

7 System Maintenance

Recommendations in this section are intended as maintenance and are intended to be checked on a frequent basis to ensure system stability. Many recommendations do not have quick remediations and require investigation into the cause and best fix available and may indicate an attempted breach of system security.

7.1 Configure system file and directory access

This section provides guidance on securing aspects of system files and directories.

7.1.1 Ensure access to /etc/passwd is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/passwd` file contains user account information that is used by many system utilities and therefore must be readable for these utilities to operate.

Rationale:

It is critical to ensure that the `/etc/passwd` file is protected from unauthorized write access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to verify `/etc/passwd` is mode 644 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)'  /etc/passwd

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to remove excess permissions, set owner, and set group on `/etc/passwd`:

```
# chmod u-x,go-wx /etc/passwd
# chown root:root /etc/passwd
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.2 Ensure access to /etc/passwd- is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/passwd-` file contains backup user account information.

Rationale:

It is critical to ensure that the `/etc/passwd-` file is protected from unauthorized access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to verify `/etc/passwd-` is mode 644 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)' /etc/passwd-

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to remove excess permissions, set owner, and set group on `/etc/passwd-`:

```
# chmod u-x,go-wx /etc/passwd-
# chown root:root /etc/passwd-
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: { 0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.3 Ensure access to /etc/group is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/group` file contains a list of all the valid groups defined in the system. The command below allows read/write access for root and read access for everyone else.

Rationale:

The `/etc/group` file needs to be protected from unauthorized changes by non-privileged users, but needs to be readable as this information is used with many non-privileged programs.

Audit:

Run the following command to verify `/etc/group` is mode 644 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)' /etc/group

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to remove excess permissions, set owner, and set group on `/etc/group`:

```
# chmod u-x,go-wx /etc/group
# chown root:root /etc/group
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.4 Ensure access to /etc/group- is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/group-` file contains a backup list of all the valid groups defined in the system.

Rationale:

It is critical to ensure that the `/etc/group-` file is protected from unauthorized access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to verify `/etc/group-` is mode 644 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)'  /etc/group-

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to remove excess permissions, set owner, and set group on `/etc/group-`:

```
# chmod u-x,go-wx /etc/group-
# chown root:root /etc/group-
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.5 Ensure access to /etc/shadow is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/shadow` file is used to store the information about user accounts that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

If attackers can gain read access to the `/etc/shadow` file, they can easily run a password cracking program against the hashed password to break it. Other security information that is stored in the `/etc/shadow` file (such as expiration) could also be useful to subvert the user accounts.

Audit:

Run the following command to verify `/etc/shadow` is mode 000, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)'  /etc/shadow

Access: (0/----------)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/shadow`:

```
# chown root:root /etc/shadow
# chmod 0000 /etc/shadow
```

Default Value:

Access: (0/----------) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.6 Ensure access to /etc/shadow- is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/shadow-` file is used to store backup information about user accounts that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

It is critical to ensure that the `/etc/shadow-` file is protected from unauthorized access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to verify `/etc/shadow-` is mode 000, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)'  /etc/shadow-

Access: (0/----------)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/shadow-`:

```
# chown root:root /etc/shadow-
# chmod 0000 /etc/shadow-
```

Default Value:

Access: (0/----------) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.7 Ensure access to /etc/gshadow is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/gshadow` file is used to store the information about groups that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

If attackers can gain read access to the `/etc/gshadow` file, they can easily run a password cracking program against the hashed password to break it. Other security information that is stored in the `/etc/gshadow` file (such as group administrators) could also be useful to subvert the group.

Audit:

Run the following command to verify `/etc/gshadow` is mode 400 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)'  /etc/gshadow

Access: (0400/-r--------)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/gshadow`:

```
# chown root:root /etc/gshadow
# chmod o-wx,go-rwx /etc/gshadow
```

Default Value:

Access: (0400/-r--------) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	16.4 Encrypt or Hash all Authentication Credentials Encrypt or hash with a salt all authentication credentials when stored.		●	●

7.1.8 Ensure access to /etc/gshadow- is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The `/etc/gshadow-` file is used to store backup information about groups that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

It is critical to ensure that the `/etc/gshadow-` file is protected from unauthorized access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to verify `/etc/gshadow-` is mode 400, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)'  /etc/gshadow-

Access: (0400/-r--------)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to set mode, owner, and group on `/etc/gshadow-`:

```
# chown root:root /etc/gshadow-
# chmod u-wx,go-rwx /etc/gshadow-
```

Default Value:

Access: (0400/-r--------) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	16.4 Encrypt or Hash all Authentication Credentials Encrypt or hash with a salt all authentication credentials when stored.		●	●

7.1.9 Ensure access to /etc/shells is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`/etc/shells` is a text file which contains the full pathnames of valid login shells. This file is consulted by `chsh` and available to be queried by other programs.

Rationale:

It is critical to ensure that the `/etc/shells` file is protected from unauthorized access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to verify `/etc/shells` is mode 644 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root`:

```
# stat -Lc 'Access: (%#a/%A)  Uid: ( %u/ %U) Gid: ( %g/ %G)' /etc/shells

Access: (0644/-rw-r--r--)  Uid: ( 0/ root) Gid: ( 0/ root)
```

Remediation:

Run the following commands to remove excess permissions, set owner, and set group on `/etc/shells`:

```
# chmod u-x,go-wx /etc/shells
# chown root:root /etc/shells
```

Default Value:

Access: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.10 Ensure access to /etc/security/opasswd is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

`/etc/security/opasswd` and it's backup `/etc/security/opasswd.old` hold user's previous passwords if `pam_unix` or `pam_pwhistory` is in use on the system

Rationale:

It is critical to ensure that `/etc/security/opasswd` is protected from unauthorized access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following commands to verify `/etc/security/opasswd` and `/etc/security/opasswd.old` are mode 600 or more restrictive, `Uid` is `0/root` and `Gid` is `0/root` if they exist:

```
#  [ -e "/etc/security/opasswd" ] && stat -Lc '%n Access: (%#a/%A)  Uid: (
%u/ %U) Gid: ( %g/ %G)' /etc/security/opasswd

/etc/security/opasswd Access: (0600/-rw-------)  Uid: ( 0/ root) Gid: ( 0/
root)
 -OR-
Nothing is returned
#  [ -e "/etc/security/opasswd.old" ] && stat -Lc '%n Access: (%#a/%A)  Uid:
( %u/ %U) Gid: ( %g/ %G)' /etc/security/opasswd.old

/etc/security/opasswd.old Access: (0600/-rw-------)  Uid: ( 0/ root) Gid: (
0/ root)
 -OR-
Nothing is returned
```

Remediation:

Run the following commands to remove excess permissions, set owner, and set group on `/etc/security/opasswd` and `/etc/security/opasswd.old` if they exist:

```
# [ -e "/etc/security/opasswd" ] && chmod u-x,go-rwx /etc/security/opasswd
# [ -e "/etc/security/opasswd" ] && chown root:root /etc/security/opasswd
# [ -e "/etc/security/opasswd.old" ] && chmod u-x,go-rwx
/etc/security/opasswd.old
# [ -e "/etc/security/opasswd.old" ] && chown root:root
/etc/security/opasswd.old
```

Default Value:

/etc/security/opasswd Access: (0600/-rw-------) Uid: (0/ root) Gid: (0/ root)

References:

1. NIST SP 800-53 Rev. 5: AC-3, MP-2

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.11 Ensure world writable files and directories are secured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

World writable files are the least secure. Data in world-writable files can be modified and compromised by any user on the system. World writable files may also indicate an incorrectly written script or program that could potentially be the cause of a larger compromise to the system's integrity. See the `chmod(2)` man page for more information.

Setting the sticky bit on world writable directories prevents users from deleting or renaming files in that directory that are not owned by them.

Rationale:

Data in world-writable files can be modified and compromised by any user on the system. World writable files may also indicate an incorrectly written script or program that could potentially be the cause of a larger compromise to the system's integrity.

This feature prevents the ability to delete or rename files in world writable directories (such as `/tmp`) that are owned by another user.

Audit:

Run the following script to verify:

- No world writable files exist
- No world writable directories without the sticky bit exist

```
#!/usr/bin/env bash

{
   l_output="" l_output2=""
   l_smask='01000'
   a_file=(); a_dir=() # Initialize arrays
   a_path=(! -path "/run/user/*" -a ! -path "/proc/*" -a ! -path
"*/containerd/*" -a ! -path "*/kubelet/pods/*" -a ! -path
"*/kubelet/plugins/*" -a ! -path "/sys/*" -a ! -path "/snap/*")
   while IFS= read -r l_mount; do
      while IFS= read -r -d $'\0' l_file; do
         if [ -e "$l_file" ]; then
            [ -f "$l_file" ] && a_file+=("$l_file") # Add WR files
            if [ -d "$l_file" ]; then # Add directories w/o sticky bit
               l_mode="$(stat -Lc '%#a' "$l_file")"
               [ ! $(( $l_mode & $l_smask )) -gt 0 ] && a_dir+=("$l_file")
            fi
         fi
      done < <(find "$l_mount" -xdev \( "${a_path[@]}" \) \( -type f -o -type
d \) -perm -0002 -print0 2> /dev/null)
   done < <(findmnt -Dkerno fstype,target | awk '($1 !~
/^\s*(nfs|proc|smb|vfat|iso9660|efivarfs|selinuxfs)/ && $2 !~
/^(\/run\/user\/|\/tmp|\/var\/tmp)/){print $2}')
   if ! (( ${#a_file[@]} > 0 )); then
      l_output="$l_output\n  - No world writable files exist on the local
filesystem."
   else
      l_output2="$l_output2\n - There are \"$(printf '%s' "${#a_file[@]}")\"
World writable files on the system.\n   - The following is a list of World
writable files:\n$(printf '%s\n' "${a_file[@]}")\n   - end of list\n"
   fi
   if ! (( ${#a_dir[@]} > 0 )); then
      l_output="$l_output\n  - Sticky bit is set on world writable
directories on the local filesystem."
   else
      l_output2="$l_output2\n - There are \"$(printf '%s' "${#a_dir[@]}")\"
World writable directories without the sticky bit on the system.\n   - The
following is a list of World writable directories without the sticky
bit:\n$(printf '%s\n' "${a_dir[@]}")\n   - end of list\n"
   fi
   unset a_path; unset a_arr; unset a_file; unset a_dir # Remove arrays
   # If l_output2 is empty, we pass
   if [ -z "$l_output2" ]; then
      echo -e "\n- Audit Result:\n  ** PASS **\n - * Correctly configured *
:\n$l_output\n"
   else
      echo -e "\n- Audit Result:\n  ** FAIL **\n - * Reasons for audit
failure * :\n$l_output2"
      [ -n "$l_output" ] && echo -e "- * Correctly configured *
:\n$l_output\n"
   fi
}
```

Note: On systems with a large number of files and/or directories, this audit may be a long running process

Remediation:

- World Writable Files:
  - It is recommended that write access is removed from `other` with the command (`chmod o-w <filename>`), but always consult relevant vendor documentation to avoid breaking any application dependencies on a given file.
- World Writable Directories:
  - Set the sticky bit on all world writable directories with the command (`chmod a+t <directory_name>`)

Run the following script to:

- Remove other write permission from any world writable files
- Add the sticky bit to all world writable directories

```
#!/usr/bin/env bash

{
   l_smask='01000'
   a_file=(); a_dir=() # Initialize arrays
   a_path=(! -path "/run/user/*" -a ! -path "/proc/*" -a ! -path
"*/containerd/*" -a ! -path "*/kubelet/pods/*" -a ! -path
"*/kubelet/plugins/*" -a ! -path "/sys/*" -a ! -path "/snap/*")
   while IFS= read -r l_mount; do
      while IFS= read -r -d $'\0' l_file; do
         if [ -e "$l_file" ]; then
            l_mode="$(stat -Lc '%#a' "$l_file")"
            if [ -f "$l_file" ]; then # Remove excess permissions from WW
files
               echo -e " - File: \"$l_file\" is mode: \"$l_mode\"\n  -
removing write permission on \"$l_file\" from \"other\""
               chmod o-w "$l_file"
            fi
            if [ -d "$l_file" ]; then # Add sticky bit
               if [ ! $(( $l_mode & $l_smask )) -gt 0 ]; then
                  echo -e " - Directory: \"$l_file\" is mode: \"$l_mode\" and
doesn't have the sticky bit set\n  - Adding the sticky bit"
                  chmod a+t "$l_file"
               fi
            fi
         fi
      done < <(find "$l_mount" -xdev \( "${a_path[@]}" \) \( -type f -o -type
d \) -perm -0002 -print0 2> /dev/null)
   done < <(findmnt -Dkerno fstype,target | awk '($1 !~
/^\s*(nfs|proc|smb|vfat|iso9660|efivarfs|selinuxfs)/ && $2 !~
/^(\/run\/user\/|\/tmp|\/var\/tmp)/){print $2}')
}
```

References:

1. NIST SP 800-53 :: SC-4
2. NIST SP 800-53A :: SC-4.1
3. RHEL 8 STIG Vul ID: V-230243
4. RHEL 8 STIG Rule ID: SV-230243r792857

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.1.12 Ensure no files or directories without an owner and a group exist (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Administrators may delete users or groups from the system and neglect to remove all files and/or directories owned by those users or groups.

Rationale:

A new user or group who is assigned a deleted user's user ID or group ID may then end up "owning" a deleted user or group's files, and thus have more access on the system than was intended.

Audit:

Run the following script to verify no unowned or ungrouped files or directories exist:

```
#!/usr/bin/env bash

{
   l_output="" l_output2=""
   a_nouser=(); a_nogroup=() # Initialize arrays
   a_path=(! -path "/run/user/*" -a ! -path "/proc/*" -a ! -path
"*/containerd/*" -a ! -path "*/kubelet/pods/*" -a ! -path
"*/kubelet/plugins/*" -a ! -path "/sys/fs/cgroup/memory/*" -a ! -path
"/var/*/private/*")
   while IFS= read -r l_mount; do
      while IFS= read -r -d $'\0' l_file; do
         if [ -e "$l_file" ]; then
            while IFS=: read -r l_user l_group; do
               [ "$l_user" = "UNKNOWN" ] && a_nouser+=("$l_file")
               [ "$l_group" = "UNKNOWN" ] && a_nogroup+=("$l_file")
            done < <(stat -Lc '%U:%G' "$l_file")
         fi
      done < <(find "$l_mount" -xdev \( "${a_path[@]}" \) \( -type f -o -type
d \) \( -nouser -o -nogroup \) -print0 2> /dev/null)
   done < <(findmnt -Dkerno fstype,target | awk '($1 !~
/^\s*(nfs|proc|smb|vfat|iso9660|efivarfs|selinuxfs)/ && $2 !~
/^\/run\/user\//){print $2}')
   if ! (( ${#a_nouser[@]} > 0 )); then
      l_output="$l_output\n  - No files or directories without a owner exist
on the local filesystem."
   else
      l_output2="$l_output2\n  - There are \"$(printf '%s'
"${#a_nouser[@]}")\" unowned files or directories on the system.\n   - The
following is a list of unowned files and/or directories:\n$(printf '%s\n'
"${a_nouser[@]}")\n   - end of list"
   fi
   if ! (( ${#a_nogroup[@]} > 0 )); then
      l_output="$l_output\n  - No files or directories without a group exist
on the local filesystem."
   else
      l_output2="$l_output2\n  - There are \"$(printf '%s'
"${#a_nogroup[@]}")\" ungrouped files or directories on the system.\n   - The
following is a list of ungrouped files and/or directories:\n$(printf '%s\n'
"${a_nogroup[@]}")\n   - end of list"
   fi
   unset a_path; unset a_arr ; unset a_nouser; unset a_nogroup # Remove
arrays
   if [ -z "$l_output2" ]; then # If l_output2 is empty, we pass
      echo -e "\n- Audit Result:\n  ** PASS **\n - * Correctly configured *
:\n$l_output\n"
   else
      echo -e "\n- Audit Result:\n  ** FAIL **\n - * Reasons for audit
failure * :\n$l_output2"
      [ -n "$l_output" ] && echo -e "\n- * Correctly configured *
:\n$l_output\n"
   fi
}
```

Note: On systems with a large number of files and/or directories, this audit may be a long running process

Remediation:

Remove or set ownership and group ownership of these files and/or directories to an active user on the system as appropriate.

References:

1. NIST SP 800-53 :: CM-6 b
2. NIST SP 800-53A :: CM-6.1 (iv)
3. RHEL 8 STIG Vul ID: V-230326
4. RHEL 8 STIG Rule ID: SV-230326r627750
5. RHEL 8 STIG Vul ID: V-230327
6. RHEL 8 STIG Rule ID: SV-230327r627750

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.2 Local User and Group Settings

This section provides guidance on securing aspects of the local users and groups.

Note: The recommendations in this section check local users and groups. Any users or groups from other sources such as LDAP will not be audited. In a domain environment similar checks should be performed against domain users and groups.

7.2.1 Ensure accounts in /etc/passwd use shadowed passwords (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Local accounts can use shadowed passwords. With shadowed passwords, the passwords are saved in shadow password file, `/etc/shadow`, encrypted by a salted one-way hash. Accounts with a shadowed password have an `x` in the second field in `/etc/passwd`.

Rationale:

The `/etc/passwd` file also contains information like user ID's and group ID's that are used by many system programs. Therefore, the `/etc/passwd` file must remain world readable. In spite of encoding the password with a randomly-generated one-way hash function, an attacker could still break the system if they got access to the `/etc/passwd` file. This can be mitigated by using shadowed passwords, thus moving the passwords in the `/etc/passwd` file to `/etc/shadow`. The `/etc/shadow` file is set so only root will be able to read and write. This helps mitigate the risk of an attacker gaining access to the encoded passwords with which to perform a dictionary attack.

Note:

- All accounts must have passwords or be locked to prevent the account from being used by an unauthorized user.
- A user account with an empty second field in `/etc/passwd` allows the account to be logged into by providing only the username.

Audit:

Run the following command and verify that no output is returned:

```
# awk -F: '($2 != "x" ) { print "User: \"" $1 "\" is not set to shadowed
passwords "}' /etc/passwd
```

Remediation:

Run the following command to set accounts to use shadowed passwords and migrate passwords in `/etc/passwd` to `/etc/shadow`:

```
# pwconv
```

Investigate to determine if the account is logged in and what it is being used for, to determine if it needs to be forced off.

References:

1. NIST SP 800-53 Rev. 5: IA-5
2. PWCONV(8)

Additional Information:

The `pwconv` command creates shadow from `passwd` and an optionally existing `shadow`.

- The `pwunconv` command creates `passwd` from `passwd` and `shadow` and then removes `shadow`.
- The `grpconv` command creates `gshadow` from `group` and an optionally existing `gshadow`.
- The `grpunconv` command creates `group` from `group` and `gshadow` and then removes `gshadow`.

These four programs all operate on the normal and shadow password and group files: `/etc/passwd`, `/etc/group`, `/etc/shadow`, and `/etc/gshadow`.

Each program acquires the necessary locks before conversion. `pwconv` and `grpconv` are similar. First, entries in the shadowed file which don't exist in the main file are removed. Then, shadowed entries which don't have `x' as the password in the main file are updated. Any missing shadowed entries are added. Finally, passwords in the main file are replaced with` x'. These programs can be used for initial conversion as well to update the shadowed file if the main file is edited by hand.

`pwconv` will use the values of `PASS_MIN_DAYS`, `PASS_MAX_DAYS`, and `PASS_WARN_AGE` from /etc/login.defs when adding new entries to /etc/shadow.

`pwunconv` and `grpunconv` are similar. Passwords in the main file are updated from the shadowed file. Entries which exist in the main file but not in the shadowed file are left alone. Finally, the shadowed file is removed. Some password aging information is lost by `pwunconv`. It will convert what it can.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	16.4 Encrypt or Hash all Authentication Credentials Encrypt or hash with a salt all authentication credentials when stored.		●	●

7.2.2 Ensure /etc/shadow password fields are not empty (Automated)

Profile Applicability:

- Level 1 - Server

Description:

An account with an empty password field means that anybody may log in as that user without providing a password.

Rationale:

All accounts must have passwords or be locked to prevent the account from being used by an unauthorized user.

Audit:

Run the following command and verify that no output is returned:

```
# awk -F: '($2 == "" ) { print $1 " does not have a password "}' /etc/shadow
```

Remediation:

If any accounts in the `/etc/shadow` file do not have a password, run the following command to lock the account until it can be determined why it does not have a password:

```
# passwd -l <username>
```

Also, check to see if the account is logged in and investigate what it is being used for to determine if it needs to be forced off.

References:

1. NIST SP 800-53 Rev. 5: IA-5
2. NIST SP 800-53 Revision 5 :: CM-6 b
3. NIST SP 800-53A :: CM-6.1 (iv)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

7.2.3 Ensure all groups in /etc/passwd exist in /etc/group (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Over time, system administration errors and changes can lead to groups being defined in `/etc/passwd` but not in `/etc/group` .

Rationale:

Groups defined in the `/etc/passwd` file but not in the `/etc/group` file pose a threat to system security since group permissions are not properly managed.

Audit:

Run the following script to verify all GIDs in `/etc/passwd` exist in `/etc/group`:

```
#!/usr/bin/env bash

{
   a_passwd_group_gid=("$(awk -F: '{print $4}' /etc/passwd | sort -u)")
   a_group_gid=("$(awk -F: '{print $3}' /etc/group | sort -u)")
   a_passwd_group_diff=("$(printf '%s\n' "${a_group_gid[@]}"
"${a_passwd_group_gid[@]}" | sort | uniq -u)")
   while IFS= read -r l_gid; do
      awk -F: '($4 == '"$l_gid"') {print "  - User: \"" $1 "\" has GID: \""
$4 "\" which does not exist in /etc/group" }' /etc/passwd
   done < <(printf '%s\n' "${a_passwd_group_gid[@]}"
"${a_passwd_group_diff[@]}" | sort | uniq -D | uniq)
   unset a_passwd_group_gid; unset a_group_gid; unset a_passwd_group_diff
}
```

Nothing should be returned

Remediation:

Analyze the output of the Audit step above and perform the appropriate action to correct any discrepancies found.

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v8	14.6 Train Workforce Members on Recognizing and Reporting Security Incidents Train workforce members to be able to recognize a potential incident and be able to report such an incident.	●	●	●

7.2.4 Ensure no duplicate UIDs exist (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Although the `useradd` program will not let you create a duplicate User ID (UID), it is possible for an administrator to manually edit the `/etc/passwd` file and change the UID field.

Rationale:

Users must be assigned unique UIDs for accountability and to ensure appropriate access protections.

Satisfies: SRG-OS-000104-GPOS-00051, SRG-OS-000121-GPOS-00062, SRG-OS-000042-GPOS-00020

Audit:

Run the following script and verify no results are returned:

```
#!/usr/bin/env bash

{
  while read -r l_count l_uid; do
    if [ "$l_count" -gt 1 ]; then
      echo -e "Duplicate UID: \"$l_uid\" Users: \"$(awk -F: '($3 == n) {
print $1 }' n=$l_uid /etc/passwd | xargs)\""
    fi
  done < <(cut -f3 -d":" /etc/passwd | sort -n | uniq -c)
}
```

Remediation:

Based on the results of the audit script, establish unique UIDs and review all files owned by the shared UIDs to determine which UID they are supposed to belong to.

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5
2. NIST SP 800-53A :: IA-2.1

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

7.2.5 Ensure no duplicate GIDs exist (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Although the `groupadd` program will not let you create a duplicate Group ID (GID), it is possible for an administrator to manually edit the `/etc/group` file and change the GID field.

Rationale:

User groups must be assigned unique GIDs for accountability and to ensure appropriate access protections.

Audit:

Run the following script and verify no results are returned:

```
#!/usr/bin/env bash

{
   while read -r l_count l_gid; do
      if [ "$l_count" -gt 1 ]; then
      echo -e "Duplicate GID: \"$l_gid\" Groups: \"$(awk -F: '($3 == n) {
print $1 }' n=$l_gid /etc/group | xargs)\""
    fi
  done < <(cut -f3 -d":" /etc/group | sort -n | uniq -c)
}
```

Remediation:

Based on the results of the audit script, establish unique GIDs and review all files owned by the shared GID to determine which group they are supposed to belong to.

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

Additional Information:

You can also use the `grpck` command to check for other inconsistencies in the `/etc/group` file.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

7.2.6 Ensure no duplicate user names exist (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Although the `useradd` program will not let you create a duplicate user name, it is possible for an administrator to manually edit the `/etc/passwd` file and change the user name.

Rationale:

If a user is assigned a duplicate user name, it will create and have access to files with the first UID for that username in `/etc/passwd` . For example, if "test4" has a UID of 1000 and a subsequent "test4" entry has a UID of 2000, logging in as "test4" will use UID 1000. Effectively, the UID is shared, which is a security problem.

Audit:

Run the following script and verify no results are returned:

```
#!/usr/bin/env bash

{
   while read -r l_count l_user; do
      if [ "$l_count" -gt 1 ]; then
         echo -e "Duplicate User: \"$l_user\" Users: \"$(awk -F: '($1 == n) {
print $1 }' n=$l_user /etc/passwd | xargs)\""
      fi
   done < <(cut -f1 -d":" /etc/passwd | sort -n | uniq -c)
}
```

Remediation:

Based on the results of the audit script, establish unique user names for the users. File ownerships will automatically reflect the change as long as the users have unique UIDs.

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

7.2.7 Ensure no duplicate group names exist (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Although the `groupadd` program will not let you create a duplicate group name, it is possible for an administrator to manually edit the `/etc/group` file and change the group name.

Rationale:

If a group is assigned a duplicate group name, it will create and have access to files with the first GID for that group in `/etc/group` . Effectively, the GID is shared, which is a security problem.

Audit:

Run the following script and verify no results are returned:

```
#!/usr/bin/env bash

{
   while read -r l_count l_group; do
      if [ "$l_count" -gt 1 ]; then
         echo -e "Duplicate Group: \"$l_group\" Groups: \"$(awk -F: '($1 ==
n) { print $1 }' n=$l_group /etc/group | xargs)\""
      fi
   done < <(cut -f1 -d":" /etc/group | sort -n | uniq -c)
}
```

Remediation:

Based on the results of the audit script, establish unique names for the user groups. File group ownerships will automatically reflect the change as long as the groups have unique GIDs.

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

7.2.8 Ensure local interactive user home directories are configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The user home directory is space defined for the particular user to set local environment variables and to store personal files. While the system administrator can establish secure permissions for users' home directories, the users can easily override these. Users can be defined in `/etc/passwd` without a home directory or with a home directory that does not actually exist.

Rationale:

Since the user is accountable for files stored in the user home directory, the user must be the owner of the directory. Group or world-writable user home directories may enable malicious users to steal or modify other users' data or to gain another user's system privileges. If the user's home directory does not exist or is unassigned, the user will be placed in "/" and will not be able to write any files or have local environment variables set.

Audit:

Run the following script to:

- Ensure local interactive user home directories exist
- Ensure local interactive users own their home directories
- Ensure local interactive user home directories are mode 750 or more restrictive

```
#!/usr/bin/env bash

{
   a_output=() a_output2=() a_exists2=() a_mode2=() a_owner2=()
   l_valid_shells="^($( awk -F\/ '$NF != "nologin" {print}' /etc/shells | sed
-rn '/^\//{s,/,\\\\/,g;p}' | paste -s -d '|' - ))$"
   l_mask='0027'; l_max="$( printf '%o' $(( 0777 & ~$l_mask)) )"
   l_users="$(awk -v pat="$l_valid_shells" -F: '$(NF) ~ pat { print $1 " "
$(NF-1) }' /etc/passwd | wc -l)"
   [ "$l_users" -gt 10000 ] && printf '%s\n' "" "  ** INFO **" \
   "  $l_users Local interactive users found on the system" "  This may be a
long running check" "  **********"
   while IFS=" " read -r l_user l_home; do
      if [ -d "$l_home" ]; then
         while IFS=: read -r l_own l_mode; do
            [ "$l_user" != "$l_own" ] && a_owner2+=("  - User: \"$l_user\"
Home \"$l_home\" is owned by: \"$l_own\"")
            [ $(( $l_mode & $l_mask )) -gt 0 ] && a_mode2+=("  - User:
\"$l_user\" Home \"$l_home\" is mode: \"$l_mode\"" \
            "    should be mode: \"$l_max\" or more restrictive")
         done <<< "$(stat -Lc '%U:%#a' "$l_home")"
      else
         a_exists2+=("  - User: \"$l_user\" Home Directory: \"$l_home\"
Doesn't exist")
      fi
   done <<< "$(awk -v pat="$l_valid_shells" -F: '$(NF) ~ pat { print $1 " "
$(NF-1) }' /etc/passwd)"
   [ "${#a_exists2[@]}" -gt 0 ] && a_output2+=("${a_exists2[@]}") || \
   a_output+=("  - All interactive users home directories exist")
   [ "${#a_mode2[@]}" -gt 0 ] && a_output2+=("${a_mode2[@]}") || \
   a_output+=("  - All interactive users home directories are mode \"$l_max\"
or more restrictive")
   [ "${#a_owner2[@]}" -gt 0 ] && a_output2+=("${a_owner2[@]}") || \
   a_output+=("  - All interactive users own their home directory")
   if [ "${#a_output2[@]}" -le 0 ]; then
      printf '%s\n' "" "- Audit Result:" "  ** PASS **" "${a_output[@]}"
   else
      printf '%s\n' "" "- Audit Result:" "  ** FAIL **" " - Reason(s) for
audit failure:" "${a_output2[@]}"
      [ "${#a_output[@]}" -gt 0 ] && printf '%s\n' "- Correctly set:"
"${a_output[@]}"
   fi
}
```

Remediation:

If a local interactive users' home directory is undefined and/or doesn't exist, follow local site policy and perform one of the following:

- Lock the user account
- Remove the user from the system
- Create a directory for the user. If undefined, edit `/etc/passwd` and add the absolute path to the directory to the last field of the user.

Run the following script to:

- Remove excessive permissions from local interactive users home directories
- Update the home directory's owner

```
#!/usr/bin/env bash

#!/usr/bin/env bash

{
   a_output=() a_output2=() a_exists2=() a_mode2=() a_owner2=()
   l_valid_shells="^($( awk -F\/ '$NF != "nologin" {print}' /etc/shells | sed
-rn '/^\//{s,/,\\\\/,g;p}' | paste -s -d '|' - ))$"
   l_mask='0027'; l_max="$( printf '%o' $(( 0777 & ~$l_mask)) )"
   l_users="$(awk -v pat="$l_valid_shells" -F: '$(NF) ~ pat { print $1 " "
$(NF-1) }' /etc/passwd | wc -l)"
   [ "$l_users" -gt 10000 ] && printf '%s\n' "" "  ** INFO **" \
   "  $l_users Local interactive users found on the system" "  This may be a
long running process" "  **********"
   while IFS=" " read -r l_user l_home; do
      if [ -d "$l_home" ]; then
         while IFS=: read -r l_own l_mode; do
            if [ "$l_user" != "$l_own" ]; then
               a_owner2+=("  - User: \"$l_user\" Home \"$l_home\" is owned
by: \"$l_own\"" \
               "    changing owner to: \"$l_user\"") && chown "$l_user"
"$l_home"
            fi
            if [ $(( $l_mode & $l_mask )) -gt 0 ]; then
               a_mode2+=("  - User: \"$l_user\" Home \"$l_home\" is mode:
\"$l_mode\"" \
               "    changing to mode: \"$l_max\" or more restrictive")
               chmod g-w,o-rwx "$l_home"
            fi
         done <<< "$(stat -Lc '%U:%#a' "$l_home")"
      else
         a_exists2+=("  - User: \"$l_user\" Home Directory: \"$l_home\"
Doesn't exist")
      fi
   done <<< "$(awk -v pat="$l_valid_shells" -F: '$(NF) ~ pat { print $1 " "
$(NF-1) }' /etc/passwd)"
   [ "${#a_exists2[@]}" -gt 0 ] && a_output2+=("${a_exists2[@]}")
   [ "${#a_mode2[@]}" -gt 0 ] && a_output2+=("${a_mode2[@]}")
   [ "${#a_owner2[@]}" -gt 0 ] && a_output2+=("${a_owner2[@]}")
   if [ "${#a_output2[@]}" -gt 0 ]; then
      printf '%s\n' "" "${a_output2[@]}"
   else
      printf '%s\n' "" "- No changes required"
   fi
}
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.2.9 Ensure local interactive user dot files access is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

While the system administrator can establish secure permissions for users' "dot" files, the users can easily override these.

- `.forward` file specifies an email address to forward the user's mail to.
- `.rhost` file provides the "remote authentication" database for the rcp, rlogin, and rsh commands and the rcmd() function. These files bypass the standard password-based user authentication mechanism. They specify remote hosts and users that are considered trusted (i.e. are allowed to access the local system without supplying a password)
- `.netrc` file contains data for logging into a remote host or passing authentication to an API.
- `.bash_history` file keeps track of the user's commands.

Rationale:

User configuration files with excessive or incorrect access may enable malicious users to steal or modify other users' data or to gain another user's system privileges.

Audit:

Run the following script to verify local interactive user dot files:

- Don't include `.forward`, `.rhost`, or `.netrc` files
- Are mode 0644 or more restrictive
- Are owned by the local interactive user
- Are group owned by the user's primary group
- `.bash_history` is mode 0600 or more restrictive

Note: If a `.netrc` file is required, and follows local site policy, it should be mode `0600` or more restrictive.

```
#!/usr/bin/env bash

{
   a_output2=(); a_output3=()
   l_maxsize="1000" # Maximum number of local interactive users before
warning (Default 1,000)
   l_valid_shells="^($( awk -F\/ '$NF != "nologin" {print}' /etc/shells | sed
-rn '/^\//{s,/,\\\\/,g;p}' | paste -s -d '|' - ))$"
   a_user_and_home=() # Create array with local users and their home
directories
   while read -r l_local_user l_local_user_home; do # Populate array with
users and user home location
      [[ -n "$l_local_user" && -n "$l_local_user_home" ]] &&
a_user_and_home+=("$l_local_user:$l_local_user_home")
   done <<< "$(awk -v pat="$l_valid_shells" -F: '$(NF) ~ pat { print $1 " "
$(NF-1) }' /etc/passwd)"
   l_asize="${#a_user_and_home[@]}" # Here if we want to look at number of
users before proceeding
   [ "${#a_user_and_home[@]}" -gt "$l_maxsize" ] && printf '%s\n' "" "  **
INFO **" \
   "  - \"$l_asize\" Local interactive users found on the system" \
   "  - This may be a long running check" ""
   file_access_chk()
   {
      a_access_out=()
      l_max="$( printf '%o' $(( 0777 & ~$l_mask)) )"
      if [ $(( $l_mode & $l_mask )) -gt 0 ]; then
         a_access_out+=("  - File: \"$l_hdfile\" is mode: \"$l_mode\" and
should be mode: \"$l_max\" or more restrictive")
      fi
      if [[ ! "$l_owner" =~ ($l_user) ]]; then
         a_access_out+=("  - File: \"$l_hdfile\" owned by: \"$l_owner\" and
should be owned by \"${l_user//|/ or }\"")
      fi
      if [[ ! "$l_gowner" =~ ($l_group) ]]; then
         a_access_out+=("  - File: \"$l_hdfile\" group owned by:
\"$l_gowner\" and should be group owned by \"${l_group//|/ or }\"")
      fi
   }
   while IFS=: read -r l_user l_home; do
      a_dot_file=(); a_netrc=(); a_netrc_warn=(); a_bhout=(); a_hdirout=()
      if [ -d "$l_home" ]; then
         l_group="$(id -gn "$l_user" | xargs)";l_group="${l_group// /|}"
         while IFS= read -r -d $'\0' l_hdfile; do
            while read -r l_mode l_owner l_gowner; do
               case "$(basename "$l_hdfile")" in
                  .forward | .rhost )
                     a_dot_file+=("  - File: \"$l_hdfile\" exists") ;;
                  .netrc )
                     l_mask='0177'; file_access_chk
                     if [ "${#a_access_out[@]}" -gt 0 ]; then
                        a_netrc+=("${a_access_out[@]}")
                     else
                        a_netrc_warn+=("   - File: \"$l_hdfile\" exists")
                     fi ;;
                  .bash_history )
                     l_mask='0177'; file_access_chk
```

```
                    [ "${#a_access_out[@]}" -gt 0 ] &&
a_bhout+=("${a_access_out[@]}") ;;
                * )
                    l_mask='0133'; file_access_chk
                    [ "${#a_access_out[@]}" -gt 0 ] &&
a_hdirout+=("${a_access_out[@]}") ;;
              esac
            done < <(stat -Lc '%#a %U %G' "$l_hdfile")
         done < <(find "$l_home" -xdev -type f -name '.*' -print0)
      fi
      if [[ "${#a_dot_file[@]}" -gt 0 || "${#a_netrc[@]}" -gt 0 ||
"${#a_bhout[@]}" -gt 0 || "${#a_hdirout[@]}" -gt 0 ]]; then
         a_output2+=(" - User: \"$l_user\" Home Directory: \"$l_home\""
"${a_dot_file[@]}" "${a_netrc[@]}" "${a_bhout[@]}" "${a_hdirout[@]}")
      fi
      [ "${#a_netrc_warn[@]}" -gt 0 ] && a_output3+=(" - User: \"$l_user\"
Home Directory: \"$l_home\"" "${a_netrc_warn[@]}")
   done <<< "$(printf '%s\n' "${a_user_and_home[@]}")"
   if [ "${#a_output2[@]}" -le 0 ]; then # If l_output2 is empty, we pass
      [ "${#a_output3[@]}" -gt 0 ] && printf '%s\n' "  ** WARNING **"
"${a_output3[@]}"
      printf '%s\n' "- Audit Result:" "  ** PASS **"
   else
      printf '%s\n' "- Audit Result:" "  ** FAIL **" " - * Reasons for audit
failure * :" "${a_output2[@]}" ""
      [ "${#a_output3[@]}" -gt 0 ] && printf '%s\n' "  ** WARNING **"
"${a_output3[@]}"
   fi
}
```

Remediation:

Making global modifications to users' files without alerting the user community can result in unexpected outages and unhappy users. Therefore, it is recommended that a monitoring policy be established to report user dot file permissions and determine the action to be taken in accordance with site policy.
The following script will:

- remove excessive permissions on `dot` files within interactive users' home directories
- change ownership of `dot` files within interactive users' home directories to the user
- change group ownership of `dot` files within interactive users' home directories to the user's primary group
- list `.forward` and `.rhost` files to be investigated and manually deleted

```
#!/usr/bin/env bash

{
   a_output2=(); a_output3=()
   l_maxsize="1000" # Maximum number of local interactive users before
warning (Default 1,000)
   l_valid_shells="^($( awk -F\/ '$NF != "nologin" {print}' /etc/shells | sed
-rn '/^\//{s,/,\\\\/,g;p}' | paste -s -d '|' - ))$"
   a_user_and_home=() # Create array with local users and their home
directories
   while read -r l_local_user l_local_user_home; do # Populate array with
users and user home location
      [[ -n "$l_local_user" && -n "$l_local_user_home" ]] &&
a_user_and_home+=("$l_local_user:$l_local_user_home")
   done <<< "$(awk -v pat="$l_valid_shells" -F: '$(NF) ~ pat { print $1 " "
$(NF-1) }' /etc/passwd)"
   l_asize="${#a_user_and_home[@]}" # Here if we want to look at number of
users before proceeding
   [ "${#a_user_and_home[@]}" -gt "$l_maxsize" ] && printf '%s\n' "" "  **
INFO **" \
   "  - \"$l_asize\" Local interactive users found on the system" \
   "  - This may be a long running check" ""
   file_access_fix()
   {
      a_access_out=()
      l_max="$( printf '%o' $(( 0777 & ~$l_mask)) )"
      if [ $(( $l_mode & $l_mask )) -gt 0 ]; then
         printf '%s\n' "" "  - File: \"$l_hdfile\" is mode: \"$l_mode\" and
should be mode: \"$l_max\" or more restrictive" \
         "     Updating file: \"$l_hdfile\" to be mode: \"$l_max\" or more
restrictive"
         chmod "$l_change" "$l_hdfile"
      fi
      if [[ ! "$l_owner" =~ ($l_user) ]]; then
         printf '%s\n' "" "  - File: \"$l_hdfile\" owned by: \"$l_owner\" and
should be owned by \"${l_user//|/ or }\"" \
         "     Updating file: \"$l_hdfile\" to be owned by \"${l_user//|/ or
}\""
         chown "$l_user" "$l_hdfile"
      fi
      if [[ ! "$l_gowner" =~ ($l_group) ]]; then
         printf '%s\n' "" "  - File: \"$l_hdfile\" group owned by:
\"$l_gowner\" and should be group owned by \"${l_group//|/ or }\"" \
         "     Updating file: \"$l_hdfile\" to be group owned by
\"${l_group//|/ or }\""
         chgrp "$l_group" "$l_hdfile"
      fi
   }
   while IFS=: read -r l_user l_home; do
      a_dot_file=(); a_netrc=(); a_netrc_warn=(); a_bhout=(); a_hdirout=()
      if [ -d "$l_home" ]; then
         l_group="$(id -gn "$l_user" | xargs)";l_group="${l_group// /|}"
         while IFS= read -r -d $'\0' l_hdfile; do
            while read -r l_mode l_owner l_gowner; do
               case "$(basename "$l_hdfile")" in
                  .forward | .rhost )
                     a_dot_file+=("  - File: \"$l_hdfile\" exists" "
```

```
Please review and manually delete this file") ;;
                  .netrc )
                     l_mask='0177'; l_change="u-x,go-rwx"; file_access_fix
                     a_netrc_warn+=("   - File: \"$l_hdfile\" exists") ;;
                  .bash_history )
                     l_mask='0177'; l_change="u-x,go-rwx"; file_access_fix ;;
                  * )
                     l_mask='0133'; l_change="u-x,go-wx"; file_access_fix ;;
               esac
            done < <(stat -Lc '%#a %U %G' "$l_hdfile")
         done < <(find "$l_home" -xdev -type f -name '.*' -print0)
      fi
      [ "${#a_dot_file[@]}" -gt 0 ] && a_output2+=(" - User: \"$l_user\" Home
Directory: \"$l_home\"" "${a_dot_file[@]}")
      [ "${#a_netrc_warn[@]}" -gt 0 ] && a_output3+=(" - User: \"$l_user\"
Home Directory: \"$l_home\"" "${a_netrc_warn[@]}")
   done <<< "$(printf '%s\n' "${a_user_and_home[@]}")"
   [ "${#a_output3[@]}" -gt 0 ] && printf '%s\n' "" "  ** WARNING **"
"${a_output3[@]}" ""
   [ "${#a_output2[@]}" -gt 0 ] && printf '%s\n' "" "${a_output2[@]}"
}
```

References:

1. NIST SP 800-53 Rev. 5: CM-1, CM-2, CM-6, CM-7, IA-5

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

Appendix: Summary Table

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1	**Initial Setup**		
1.1	**Filesystem**		
1.1.1	**Configure Filesystem Kernel Modules**		
1.1.1.1	Ensure cramfs kernel module is not available (Automated)	☐	☐
1.1.1.2	Ensure freevxfs kernel module is not available (Automated)	☐	☐
1.1.1.3	Ensure hfs kernel module is not available (Automated)	☐	☐
1.1.1.4	Ensure hfsplus kernel module is not available (Automated)	☐	☐
1.1.1.5	Ensure jffs2 kernel module is not available (Automated)	☐	☐
1.1.1.6	Ensure unused filesystems kernel modules are not available (Manual)	☐	☐
1.1.2	**Configure Filesystem Partitions**		
1.1.2.1	**Configure /tmp**		
1.1.2.1.1	Ensure /tmp is a separate partition (Automated)	☐	☐
1.1.2.1.2	Ensure nodev option set on /tmp partition (Automated)	☐	☐
1.1.2.1.3	Ensure nosuid option set on /tmp partition (Automated)	☐	☐
1.1.2.2	**Configure /dev/shm**		
1.1.2.2.1	Ensure /dev/shm is a separate partition (Automated)	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition (Automated)	☐	☐
1.2	**Package Management**		
1.2.1	**Configure Package Repositories**		
1.2.1.1	Ensure GPG keys are configured (Manual)	☐	☐
1.2.1.2	Ensure gpgcheck is configured (Automated)	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated (Automated)	☐	☐
1.2.1.4	Ensure package manager repositories are configured (Manual)	☐	☐
1.3	**Configure Additional Process Hardening**		
1.3.1	Ensure address space layout randomization is enabled (Automated)	☐	☐
1.3.2	Ensure ptrace_scope is restricted (Automated)	☐	☐
1.3.3	Ensure core dump backtraces are disabled (Automated)	☐	☐
1.3.4	Ensure core dump storage is disabled (Automated)	☐	☐
1.4	**Configure Command Line Warning Banners**		
1.4.1	Ensure local login warning banner is configured properly (Automated)	☐	☐
1.4.2	Ensure remote login warning banner is configured properly (Automated)	☐	☐
1.4.3	Ensure access to /etc/motd is configured (Automated)	☐	☐
1.4.4	Ensure access to /etc/issue is configured (Automated)	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
2	**Services**		
2.1	**Configure Time Synchronization**		
2.1.1	Ensure time synchronization is in use (Automated)	☐	☐
2.1.2	Ensure chrony is configured (Automated)	☐	☐
2.2	**Configure Special Purpose Services**		
2.2.1	Ensure xinetd is not installed (Automated)	☐	☐
2.2.2	Ensure xorg-x11-server-common is not installed (Automated)	☐	☐
2.2.3	Ensure avahi is not installed (Automated)	☐	☐
2.2.4	Ensure a print server is not installed (Automated)	☐	☐
2.2.5	Ensure a dhcp server is not installed (Automated)	☐	☐
2.2.6	Ensure a dns server is not installed (Automated)	☐	☐
2.2.7	Ensure FTP client is not installed (Automated)	☐	☐
2.2.8	Ensure an ftp server is not installed (Automated)	☐	☐
2.2.9	Ensure a tftp server is not installed (Automated)	☐	☐
2.2.10	Ensure a web server is not installed (Automated)	☐	☐
2.2.11	Ensure IMAP and POP3 server is not installed (Automated)	☐	☐
2.2.12	Ensure Samba is not installed (Automated)	☐	☐
2.2.13	Ensure HTTP Proxy Server is not installed (Automated)	☐	☐
2.2.14	Ensure net-snmp is not installed or the snmpd service is not enabled (Automated)	☐	☐
2.2.15	Ensure NIS server is not installed (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
2.2.16	Ensure telnet-server is not installed (Automated)	☐	☐
2.2.17	Ensure mail transfer agent is configured for local-only mode (Automated)	☐	☐
2.2.18	Ensure nfs-utils is not installed or the nfs-server service is masked (Automated)	☐	☐
2.2.19	Ensure rsync-daemon is not installed or the rsyncd service is masked (Automated)	☐	☐
2.3	**Service Clients**		
2.3.1	Ensure NIS Client is not installed (Automated)	☐	☐
2.3.2	Ensure rsh client is not installed (Automated)	☐	☐
2.3.3	Ensure talk client is not installed (Automated)	☐	☐
2.3.4	Ensure telnet client is not installed (Automated)	☐	☐
2.3.5	Ensure LDAP client is not installed (Automated)	☐	☐
2.3.6	Ensure TFTP client is not installed (Automated)	☐	☐
3	**Network**		
3.1	**Configure Network Kernel Parameters**		
3.1.1	Ensure packet redirect sending is disabled (Automated)	☐	☐
3.1.2	Ensure bogus icmp responses are ignored (Automated)	☐	☐
3.1.3	Ensure broadcast icmp requests are ignored (Automated)	☐	☐
3.1.4	Ensure icmp redirects are not accepted (Automated)	☐	☐
3.1.5	Ensure secure icmp redirects are not accepted (Automated)	☐	☐
3.1.6	Ensure reverse path filtering is enabled (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
3.1.7	Ensure source routed packets are not accepted (Automated)	☐	☐
3.1.8	Ensure suspicious packets are logged (Automated)	☐	☐
3.1.9	Ensure tcp syn cookies is enabled (Automated)	☐	☐
3.1.10	Ensure ipv6 router advertisements are not accepted (Automated)	☐	☐
4	**Host Based Firewall**		
4.1	**Configure host based firewall packages**		
4.1.1	Ensure iptables is installed (Automated)	☐	☐
4.1.2	Ensure nftables is not in use (Automated)	☐	☐
4.1.3	Ensure firewalld is not in use (Automated)	☐	☐
5	**Access, Authentication and Authorization**		
5.1	**Configure time-based job schedulers**		
5.1.1	Ensure cron daemon is enabled (Automated)	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured (Automated)	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured (Automated)	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured (Automated)	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured (Automated)	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
5.1.7	Ensure permissions on /etc/cron.d are configured (Automated)	☐	☐
5.1.8	Ensure cron is restricted to authorized users (Automated)	☐	☐
5.1.9	Ensure at is restricted to authorized users (Automated)	☐	☐
5.2	**Configure SSH Server**		
5.2.1	Ensure access to /etc/ssh/sshd_config is configured (Automated)	☐	☐
5.2.2	Ensure access to SSH private host key files is configured (Automated)	☐	☐
5.2.3	Ensure access to SSH public host key files is configured (Automated)	☐	☐
5.2.4	Ensure sshd Ciphers are configured (Automated)	☐	☐
5.2.5	Ensure sshd KexAlgorithms is configured (Automated)	☐	☐
5.2.6	Ensure sshd MACs are configured (Automated)	☐	☐
5.2.7	Ensure sshd access is configured (Automated)	☐	☐
5.2.8	Ensure sshd Banner is configured (Automated)	☐	☐
5.2.9	Ensure sshd ClientAliveInterval and ClientAliveCountMax are configured (Automated)	☐	☐
5.2.10	Ensure sshd HostbasedAuthentication is disabled (Automated)	☐	☐
5.2.11	Ensure sshd IgnoreRhosts is enabled (Automated)	☐	☐
5.2.12	Ensure sshd LoginGraceTime is configured (Automated)	☐	☐
5.2.13	Ensure sshd LogLevel is configured (Automated)	☐	☐
5.2.14	Ensure sshd MaxAuthTries is configured (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
5.2.15	Ensure sshd MaxStartups is configured (Automated)	☐	☐
5.2.16	Ensure sshd MaxSessions is configured (Automated)	☐	☐
5.2.17	Ensure sshd PermitEmptyPasswords is disabled (Automated)	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled (Automated)	☐	☐
5.2.19	Ensure sshd PermitUserEnvironment is disabled (Automated)	☐	☐
5.2.20	Ensure sshd UsePAM is enabled (Automated)	☐	☐
5.3	**Configure privilege escalation**		
5.3.1	Ensure sudo is installed (Automated)	☐	☐
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally (Automated)	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly (Automated)	☐	☐
5.4	**Configure PAM**		
5.4.1	Ensure password creation requirements are configured (Automated)	☐	☐
5.4.2	Ensure lockout for failed password attempts is configured (Automated)	☐	☐
5.4.3	Ensure password hashing algorithm is SHA-512 (Automated)	☐	☐
5.4.4	Ensure password reuse is limited (Automated)	☐	☐
5.5	**User Accounts and Environment**		
5.5.1	**Set Shadow Password Suite Parameters**		

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
5.5.1.1	Ensure password expiration is 365 days or less (Automated)	☐	☐
5.5.1.2	Ensure minimum days between password changes is configured (Automated)	☐	☐
5.5.1.3	Ensure password expiration warning days is 7 or more (Automated)	☐	☐
5.5.1.4	Ensure inactive password lock is 30 days or less (Automated)	☐	☐
5.5.1.5	Ensure all users last password change date is in the past (Automated)	☐	☐
5.5.2	Ensure system accounts are secured (Automated)	☐	☐
5.5.3	Ensure default group for the root account is GID 0 (Automated)	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive (Automated)	☐	☐
6	**Logging and Auditing**		
6.1	**System Logging**		
6.1.1	**Configure journald**		
6.1.1.1	**Configure systemd-journald service**		
6.1.1.1.1	Ensure journald service is active (Automated)	☐	☐
6.1.1.1.2	Ensure journald log file access is configured (Manual)	☐	☐
6.1.1.1.3	Ensure journald ForwardToSyslog is configured (Automated)	☐	☐
6.1.1.1.4	Ensure systemd-journal-remote service is not in use (Automated)	☐	☐
6.1.1.1.5	Ensure journald Storage is configured (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
6.1.1.1.6	Ensure journald Compress is configured (Automated)	☐	☐
6.1.2	**Configure rsyslog**		
6.1.2.1	Ensure rsyslog service is enabled and active (Automated)	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured (Automated)	☐	☐
6.1.2.3	Ensure rsyslog is not configured to receive logs from a remote client (Automated)	☐	☐
6.1.3	**Configure Logfiles**		
6.1.3.1	Ensure access to all logfiles has been configured (Automated)	☐	☐
6.2	Ensure logrotate is configured (Manual)	☐	☐
7	**System Maintenance**		
7.1	**Configure system file and directory access**		
7.1.1	Ensure access to /etc/passwd is configured (Automated)	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured (Automated)	☐	☐
7.1.3	Ensure access to /etc/group is configured (Automated)	☐	☐
7.1.4	Ensure access to /etc/group- is configured (Automated)	☐	☐
7.1.5	Ensure access to /etc/shadow is configured (Automated)	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured (Automated)	☐	☐
7.1.7	Ensure access to /etc/gshadow is configured (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
7.1.8	Ensure access to /etc/gshadow- is configured (Automated)	☐	☐
7.1.9	Ensure access to /etc/shells is configured (Automated)	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured (Automated)	☐	☐
7.1.11	Ensure world writable files and directories are secured (Automated)	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist (Automated)	☐	☐
7.2	**Local User and Group Settings**		
7.2.1	Ensure accounts in /etc/passwd use shadowed passwords (Automated)	☐	☐
7.2.2	Ensure /etc/shadow password fields are not empty (Automated)	☐	☐
7.2.3	Ensure all groups in /etc/passwd exist in /etc/group (Automated)	☐	☐
7.2.4	Ensure no duplicate UIDs exist (Automated)	☐	☐
7.2.5	Ensure no duplicate GIDs exist (Automated)	☐	☐
7.2.6	Ensure no duplicate user names exist (Automated)	☐	☐
7.2.7	Ensure no duplicate group names exist (Automated)	☐	☐
7.2.8	Ensure local interactive user home directories are configured (Automated)	☐	☐
7.2.9	Ensure local interactive user dot files access is configured (Automated)	☐	☐

Appendix: CIS Controls v7 IG 1 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.2.1.3	Ensure nosuid option set on /tmp partition	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition	☐	☐
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition	☐	☐
1.2.1.1	Ensure GPG keys are configured	☐	☐
1.2.1.2	Ensure gpgcheck is configured	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated	☐	☐
1.2.1.4	Ensure package manager repositories are configured	☐	☐
1.4.3	Ensure access to /etc/motd is configured	☐	☐
1.4.4	Ensure access to /etc/issue is configured	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured	☐	☐
2.2.1	Ensure xinetd is not installed	☐	☐
2.2.15	Ensure NIS server is not installed	☐	☐
2.2.16	Ensure telnet-server is not installed	☐	☐
2.3.1	Ensure NIS Client is not installed	☐	☐
2.3.2	Ensure rsh client is not installed	☐	☐
2.3.3	Ensure talk client is not installed	☐	☐
2.3.4	Ensure telnet client is not installed	☐	☐
2.3.5	Ensure LDAP client is not installed	☐	☐
3.1.8	Ensure suspicious packets are logged	☐	☐
4.1.1	Ensure iptables is installed	☐	☐
4.1.2	Ensure nftables is not in use	☐	☐
4.1.3	Ensure firewalld is not in use	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
5.1.7	Ensure permissions on /etc/cron.d are configured	☐	☐
5.1.8	Ensure cron is restricted to authorized users	☐	☐
5.1.9	Ensure at is restricted to authorized users	☐	☐
5.2.1	Ensure access to /etc/ssh/sshd_config is configured	☐	☐
5.2.2	Ensure access to SSH private host key files is configured	☐	☐
5.2.3	Ensure access to SSH public host key files is configured	☐	☐
5.2.7	Ensure sshd access is configured	☐	☐
5.2.13	Ensure sshd LogLevel is configured	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled	☐	☐
5.3.1	Ensure sudo is installed	☐	☐
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly	☐	☐
5.5.2	Ensure system accounts are secured	☐	☐
5.5.3	Ensure default group for the root account is GID 0	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive	☐	☐
6.1.1.1.1	Ensure journald service is active	☐	☐
6.1.1.1.2	Ensure journald log file access is configured	☐	☐
6.1.1.1.5	Ensure journald Storage is configured	☐	☐
6.1.1.1.6	Ensure journald Compress is configured	☐	☐
6.1.2.1	Ensure rsyslog service is enabled and active	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured	☐	☐
6.1.3.1	Ensure access to all logfiles has been configured	☐	☐
7.1.1	Ensure access to /etc/passwd is configured	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured	☐	☐
7.1.3	Ensure access to /etc/group is configured	☐	☐
7.1.4	Ensure access to /etc/group- is configured	☐	☐
7.1.5	Ensure access to /etc/shadow is configured	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured	☐	☐
7.1.9	Ensure access to /etc/shells is configured	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
7.1.11	Ensure world writable files and directories are secured	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist	☐	☐
7.2.8	Ensure local interactive user home directories are configured	☐	☐
7.2.9	Ensure local interactive user dot files access is configured	☐	☐

Appendix: CIS Controls v7 IG 2 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1.1	Ensure cramfs kernel module is not available	☐	☐
1.1.1.2	Ensure freevxfs kernel module is not available	☐	☐
1.1.1.3	Ensure hfs kernel module is not available	☐	☐
1.1.1.4	Ensure hfsplus kernel module is not available	☐	☐
1.1.1.5	Ensure jffs2 kernel module is not available	☐	☐
1.1.1.6	Ensure unused filesystems kernel modules are not available	☐	☐
1.1.2.1.1	Ensure /tmp is a separate partition	☐	☐
1.1.2.1.2	Ensure nodev option set on /tmp partition	☐	☐
1.1.2.1.3	Ensure nosuid option set on /tmp partition	☐	☐
1.1.2.2.1	Ensure /dev/shm is a separate partition	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition	☐	☐
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition	☐	☐
1.2.1.1	Ensure GPG keys are configured	☐	☐
1.2.1.2	Ensure gpgcheck is configured	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated	☐	☐
1.2.1.4	Ensure package manager repositories are configured	☐	☐
1.3.1	Ensure address space layout randomization is enabled	☐	☐
1.3.2	Ensure ptrace_scope is restricted	☐	☐
1.4.3	Ensure access to /etc/motd is configured	☐	☐
1.4.4	Ensure access to /etc/issue is configured	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured	☐	☐
2.1.1	Ensure time synchronization is in use	☐	☐
2.1.2	Ensure chrony is configured	☐	☐
2.2.1	Ensure xinetd is not installed	☐	☐
2.2.2	Ensure xorg-x11-server-common is not installed	☐	☐
2.2.3	Ensure avahi is not installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
2.2.4	Ensure a print server is not installed	☐	☐
2.2.5	Ensure a dhcp server is not installed	☐	☐
2.2.6	Ensure a dns server is not installed	☐	☐
2.2.7	Ensure FTP client is not installed	☐	☐
2.2.8	Ensure an ftp server is not installed	☐	☐
2.2.9	Ensure a tftp server is not installed	☐	☐
2.2.10	Ensure a web server is not installed	☐	☐
2.2.11	Ensure IMAP and POP3 server is not installed	☐	☐
2.2.12	Ensure Samba is not installed	☐	☐
2.2.13	Ensure HTTP Proxy Server is not installed	☐	☐
2.2.14	Ensure net-snmp is not installed or the snmpd service is not enabled	☐	☐
2.2.15	Ensure NIS server is not installed	☐	☐
2.2.16	Ensure telnet-server is not installed	☐	☐
2.2.17	Ensure mail transfer agent is configured for local-only mode	☐	☐
2.2.18	Ensure nfs-utils is not installed or the nfs-server service is masked	☐	☐
2.2.19	Ensure rsync-daemon is not installed or the rsyncd service is masked	☐	☐
2.3.1	Ensure NIS Client is not installed	☐	☐
2.3.2	Ensure rsh client is not installed	☐	☐
2.3.3	Ensure talk client is not installed	☐	☐
2.3.4	Ensure telnet client is not installed	☐	☐
2.3.5	Ensure LDAP client is not installed	☐	☐
2.3.6	Ensure TFTP client is not installed	☐	☐
3.1.1	Ensure packet redirect sending is disabled	☐	☐
3.1.2	Ensure bogus icmp responses are ignored	☐	☐
3.1.3	Ensure broadcast icmp requests are ignored	☐	☐
3.1.4	Ensure icmp redirects are not accepted	☐	☐
3.1.5	Ensure secure icmp redirects are not accepted	☐	☐
3.1.6	Ensure reverse path filtering is enabled	☐	☐

Recommendation		Set Correctly	
		Yes	No
3.1.7	Ensure source routed packets are not accepted	☐	☐
3.1.8	Ensure suspicious packets are logged	☐	☐
3.1.9	Ensure tcp syn cookies is enabled	☐	☐
3.1.10	Ensure ipv6 router advertisements are not accepted	☐	☐
4.1.1	Ensure iptables is installed	☐	☐
4.1.2	Ensure nftables is not in use	☐	☐
4.1.3	Ensure firewalld is not in use	☐	☐
5.1.1	Ensure cron daemon is enabled	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured	☐	☐
5.1.7	Ensure permissions on /etc/cron.d are configured	☐	☐
5.1.8	Ensure cron is restricted to authorized users	☐	☐
5.1.9	Ensure at is restricted to authorized users	☐	☐
5.2.1	Ensure access to /etc/ssh/sshd_config is configured	☐	☐
5.2.2	Ensure access to SSH private host key files is configured	☐	☐
5.2.3	Ensure access to SSH public host key files is configured	☐	☐
5.2.4	Ensure sshd Ciphers are configured	☐	☐
5.2.5	Ensure sshd KexAlgorithms is configured	☐	☐
5.2.6	Ensure sshd MACs are configured	☐	☐
5.2.7	Ensure sshd access is configured	☐	☐
5.2.10	Ensure sshd HostbasedAuthentication is disabled	☐	☐
5.2.11	Ensure sshd IgnoreRhosts is enabled	☐	☐
5.2.13	Ensure sshd LogLevel is configured	☐	☐
5.2.17	Ensure sshd PermitEmptyPasswords is disabled	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled	☐	☐
5.2.20	Ensure sshd UsePAM is enabled	☐	☐
5.3.1	Ensure sudo is installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly	☐	☐
5.4.1	Ensure password creation requirements are configured	☐	☐
5.4.2	Ensure lockout for failed password attempts is configured	☐	☐
5.4.3	Ensure password hashing algorithm is SHA-512	☐	☐
5.5.1.1	Ensure password expiration is 365 days or less	☐	☐
5.5.1.2	Ensure minimum days between password changes is configured	☐	☐
5.5.1.3	Ensure password expiration warning days is 7 or more	☐	☐
5.5.1.4	Ensure inactive password lock is 30 days or less	☐	☐
5.5.1.5	Ensure all users last password change date is in the past	☐	☐
5.5.2	Ensure system accounts are secured	☐	☐
5.5.3	Ensure default group for the root account is GID 0	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive	☐	☐
6.1.1.1.1	Ensure journald service is active	☐	☐
6.1.1.1.2	Ensure journald log file access is configured	☐	☐
6.1.1.1.3	Ensure journald ForwardToSyslog is configured	☐	☐
6.1.1.1.4	Ensure systemd-journal-remote service is not in use	☐	☐
6.1.1.1.5	Ensure journald Storage is configured	☐	☐
6.1.1.1.6	Ensure journald Compress is configured	☐	☐
6.1.2.1	Ensure rsyslog service is enabled and active	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured	☐	☐
6.1.2.3	Ensure rsyslog is not configured to receive logs from a remote client	☐	☐
6.1.3.1	Ensure access to all logfiles has been configured	☐	☐
6.2	Ensure logrotate is configured	☐	☐
7.1.1	Ensure access to /etc/passwd is configured	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured	☐	☐
7.1.3	Ensure access to /etc/group is configured	☐	☐
7.1.4	Ensure access to /etc/group- is configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
7.1.5	Ensure access to /etc/shadow is configured	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured	☐	☐
7.1.7	Ensure access to /etc/gshadow is configured	☐	☐
7.1.8	Ensure access to /etc/gshadow- is configured	☐	☐
7.1.9	Ensure access to /etc/shells is configured	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured	☐	☐
7.1.11	Ensure world writable files and directories are secured	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist	☐	☐
7.2.1	Ensure accounts in /etc/passwd use shadowed passwords	☐	☐
7.2.2	Ensure /etc/shadow password fields are not empty	☐	☐
7.2.8	Ensure local interactive user home directories are configured	☐	☐
7.2.9	Ensure local interactive user dot files access is configured	☐	☐

Appendix: CIS Controls v7 IG 3 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1.1	Ensure cramfs kernel module is not available	☐	☐
1.1.1.2	Ensure freevxfs kernel module is not available	☐	☐
1.1.1.3	Ensure hfs kernel module is not available	☐	☐
1.1.1.4	Ensure hfsplus kernel module is not available	☐	☐
1.1.1.5	Ensure jffs2 kernel module is not available	☐	☐
1.1.1.6	Ensure unused filesystems kernel modules are not available	☐	☐
1.1.2.1.1	Ensure /tmp is a separate partition	☐	☐
1.1.2.1.2	Ensure nodev option set on /tmp partition	☐	☐
1.1.2.1.3	Ensure nosuid option set on /tmp partition	☐	☐
1.1.2.2.1	Ensure /dev/shm is a separate partition	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition	☐	☐
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition	☐	☐
1.2.1.1	Ensure GPG keys are configured	☐	☐
1.2.1.2	Ensure gpgcheck is configured	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated	☐	☐
1.2.1.4	Ensure package manager repositories are configured	☐	☐
1.3.1	Ensure address space layout randomization is enabled	☐	☐
1.3.2	Ensure ptrace_scope is restricted	☐	☐
1.4.3	Ensure access to /etc/motd is configured	☐	☐
1.4.4	Ensure access to /etc/issue is configured	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured	☐	☐
2.1.1	Ensure time synchronization is in use	☐	☐
2.1.2	Ensure chrony is configured	☐	☐
2.2.1	Ensure xinetd is not installed	☐	☐
2.2.2	Ensure xorg-x11-server-common is not installed	☐	☐
2.2.3	Ensure avahi is not installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
2.2.4	Ensure a print server is not installed	☐	☐
2.2.5	Ensure a dhcp server is not installed	☐	☐
2.2.6	Ensure a dns server is not installed	☐	☐
2.2.7	Ensure FTP client is not installed	☐	☐
2.2.8	Ensure an ftp server is not installed	☐	☐
2.2.9	Ensure a tftp server is not installed	☐	☐
2.2.10	Ensure a web server is not installed	☐	☐
2.2.11	Ensure IMAP and POP3 server is not installed	☐	☐
2.2.12	Ensure Samba is not installed	☐	☐
2.2.13	Ensure HTTP Proxy Server is not installed	☐	☐
2.2.14	Ensure net-snmp is not installed or the snmpd service is not enabled	☐	☐
2.2.15	Ensure NIS server is not installed	☐	☐
2.2.16	Ensure telnet-server is not installed	☐	☐
2.2.17	Ensure mail transfer agent is configured for local-only mode	☐	☐
2.2.18	Ensure nfs-utils is not installed or the nfs-server service is masked	☐	☐
2.2.19	Ensure rsync-daemon is not installed or the rsyncd service is masked	☐	☐
2.3.1	Ensure NIS Client is not installed	☐	☐
2.3.2	Ensure rsh client is not installed	☐	☐
2.3.3	Ensure talk client is not installed	☐	☐
2.3.4	Ensure telnet client is not installed	☐	☐
2.3.5	Ensure LDAP client is not installed	☐	☐
2.3.6	Ensure TFTP client is not installed	☐	☐
3.1.1	Ensure packet redirect sending is disabled	☐	☐
3.1.2	Ensure bogus icmp responses are ignored	☐	☐
3.1.3	Ensure broadcast icmp requests are ignored	☐	☐
3.1.4	Ensure icmp redirects are not accepted	☐	☐
3.1.5	Ensure secure icmp redirects are not accepted	☐	☐
3.1.6	Ensure reverse path filtering is enabled	☐	☐

Recommendation		Set Correctly	
		Yes	No
3.1.7	Ensure source routed packets are not accepted	☐	☐
3.1.8	Ensure suspicious packets are logged	☐	☐
3.1.9	Ensure tcp syn cookies is enabled	☐	☐
3.1.10	Ensure ipv6 router advertisements are not accepted	☐	☐
4.1.1	Ensure iptables is installed	☐	☐
4.1.2	Ensure nftables is not in use	☐	☐
4.1.3	Ensure firewalld is not in use	☐	☐
5.1.1	Ensure cron daemon is enabled	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured	☐	☐
5.1.7	Ensure permissions on /etc/cron.d are configured	☐	☐
5.1.8	Ensure cron is restricted to authorized users	☐	☐
5.1.9	Ensure at is restricted to authorized users	☐	☐
5.2.1	Ensure access to /etc/ssh/sshd_config is configured	☐	☐
5.2.2	Ensure access to SSH private host key files is configured	☐	☐
5.2.3	Ensure access to SSH public host key files is configured	☐	☐
5.2.4	Ensure sshd Ciphers are configured	☐	☐
5.2.5	Ensure sshd KexAlgorithms is configured	☐	☐
5.2.6	Ensure sshd MACs are configured	☐	☐
5.2.7	Ensure sshd access is configured	☐	☐
5.2.10	Ensure sshd HostbasedAuthentication is disabled	☐	☐
5.2.11	Ensure sshd IgnoreRhosts is enabled	☐	☐
5.2.13	Ensure sshd LogLevel is configured	☐	☐
5.2.14	Ensure sshd MaxAuthTries is configured	☐	☐
5.2.17	Ensure sshd PermitEmptyPasswords is disabled	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled	☐	☐
5.2.20	Ensure sshd UsePAM is enabled	☐	☐
5.3.1	Ensure sudo is installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly	☐	☐
5.4.1	Ensure password creation requirements are configured	☐	☐
5.4.2	Ensure lockout for failed password attempts is configured	☐	☐
5.4.3	Ensure password hashing algorithm is SHA-512	☐	☐
5.5.1.1	Ensure password expiration is 365 days or less	☐	☐
5.5.1.2	Ensure minimum days between password changes is configured	☐	☐
5.5.1.3	Ensure password expiration warning days is 7 or more	☐	☐
5.5.1.4	Ensure inactive password lock is 30 days or less	☐	☐
5.5.1.5	Ensure all users last password change date is in the past	☐	☐
5.5.2	Ensure system accounts are secured	☐	☐
5.5.3	Ensure default group for the root account is GID 0	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive	☐	☐
6.1.1.1.1	Ensure journald service is active	☐	☐
6.1.1.1.2	Ensure journald log file access is configured	☐	☐
6.1.1.1.3	Ensure journald ForwardToSyslog is configured	☐	☐
6.1.1.1.4	Ensure systemd-journal-remote service is not in use	☐	☐
6.1.1.1.5	Ensure journald Storage is configured	☐	☐
6.1.1.1.6	Ensure journald Compress is configured	☐	☐
6.1.2.1	Ensure rsyslog service is enabled and active	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured	☐	☐
6.1.2.3	Ensure rsyslog is not configured to receive logs from a remote client	☐	☐
6.1.3.1	Ensure access to all logfiles has been configured	☐	☐
6.2	Ensure logrotate is configured	☐	☐
7.1.1	Ensure access to /etc/passwd is configured	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured	☐	☐
7.1.3	Ensure access to /etc/group is configured	☐	☐
7.1.4	Ensure access to /etc/group- is configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
7.1.5	Ensure access to /etc/shadow is configured	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured	☐	☐
7.1.7	Ensure access to /etc/gshadow is configured	☐	☐
7.1.8	Ensure access to /etc/gshadow- is configured	☐	☐
7.1.9	Ensure access to /etc/shells is configured	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured	☐	☐
7.1.11	Ensure world writable files and directories are secured	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist	☐	☐
7.2.1	Ensure accounts in /etc/passwd use shadowed passwords	☐	☐
7.2.2	Ensure /etc/shadow password fields are not empty	☐	☐
7.2.8	Ensure local interactive user home directories are configured	☐	☐
7.2.9	Ensure local interactive user dot files access is configured	☐	☐

Appendix: CIS Controls v7 Unmapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.3.3	Ensure core dump backtraces are disabled	☐	☐
1.3.4	Ensure core dump storage is disabled	☐	☐
1.4.1	Ensure local login warning banner is configured properly	☐	☐
1.4.2	Ensure remote login warning banner is configured properly	☐	☐
7.2.3	Ensure all groups in /etc/passwd exist in /etc/group	☐	☐

Appendix: CIS Controls v8 IG 1 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.2.1.3	Ensure nosuid option set on /tmp partition	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition	☐	☐
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition	☐	☐
1.2.1.1	Ensure GPG keys are configured	☐	☐
1.2.1.2	Ensure gpgcheck is configured	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated	☐	☐
1.2.1.4	Ensure package manager repositories are configured	☐	☐
1.4.3	Ensure access to /etc/motd is configured	☐	☐
1.4.4	Ensure access to /etc/issue is configured	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured	☐	☐
4.1.1	Ensure iptables is installed	☐	☐
4.1.2	Ensure nftables is not in use	☐	☐
4.1.3	Ensure firewalld is not in use	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured	☐	☐
5.1.7	Ensure permissions on /etc/cron.d are configured	☐	☐
5.1.8	Ensure cron is restricted to authorized users	☐	☐
5.1.9	Ensure at is restricted to authorized users	☐	☐
5.2.1	Ensure access to /etc/ssh/sshd_config is configured	☐	☐
5.2.2	Ensure access to SSH private host key files is configured	☐	☐
5.2.3	Ensure access to SSH public host key files is configured	☐	☐
5.2.7	Ensure sshd access is configured	☐	☐
5.2.10	Ensure sshd HostbasedAuthentication is disabled	☐	☐
5.2.11	Ensure sshd IgnoreRhosts is enabled	☐	☐

Recommendation		Set Correctly	
		Yes	No
5.2.13	Ensure sshd LogLevel is configured	☐	☐
5.2.17	Ensure sshd PermitEmptyPasswords is disabled	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled	☐	☐
5.2.20	Ensure sshd UsePAM is enabled	☐	☐
5.3.1	Ensure sudo is installed	☐	☐
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly	☐	☐
5.4.1	Ensure password creation requirements are configured	☐	☐
5.4.2	Ensure lockout for failed password attempts is configured	☐	☐
5.4.4	Ensure password reuse is limited	☐	☐
5.5.1.1	Ensure password expiration is 365 days or less	☐	☐
5.5.1.2	Ensure minimum days between password changes is configured	☐	☐
5.5.1.3	Ensure password expiration warning days is 7 or more	☐	☐
5.5.1.4	Ensure inactive password lock is 30 days or less	☐	☐
5.5.1.5	Ensure all users last password change date is in the past	☐	☐
5.5.2	Ensure system accounts are secured	☐	☐
5.5.3	Ensure default group for the root account is GID 0	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive	☐	☐
6.1.1.1.1	Ensure journald service is active	☐	☐
6.1.1.1.2	Ensure journald log file access is configured	☐	☐
6.1.1.1.3	Ensure journald ForwardToSyslog is configured	☐	☐
6.1.1.1.5	Ensure journald Storage is configured	☐	☐
6.1.1.1.6	Ensure journald Compress is configured	☐	☐
6.1.2.1	Ensure rsyslog service is enabled and active	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured	☐	☐
6.1.3.1	Ensure access to all logfiles has been configured	☐	☐
6.2	Ensure logrotate is configured	☐	☐
7.1.1	Ensure access to /etc/passwd is configured	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
7.1.3	Ensure access to /etc/group is configured	☐	☐
7.1.4	Ensure access to /etc/group- is configured	☐	☐
7.1.5	Ensure access to /etc/shadow is configured	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured	☐	☐
7.1.7	Ensure access to /etc/gshadow is configured	☐	☐
7.1.8	Ensure access to /etc/gshadow- is configured	☐	☐
7.1.9	Ensure access to /etc/shells is configured	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured	☐	☐
7.1.11	Ensure world writable files and directories are secured	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist	☐	☐
7.2.2	Ensure /etc/shadow password fields are not empty	☐	☐
7.2.3	Ensure all groups in /etc/passwd exist in /etc/group	☐	☐
7.2.8	Ensure local interactive user home directories are configured	☐	☐
7.2.9	Ensure local interactive user dot files access is configured	☐	☐

Appendix: CIS Controls v8 IG 2 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1.1	Ensure cramfs kernel module is not available	☐	☐
1.1.1.2	Ensure freevxfs kernel module is not available	☐	☐
1.1.1.3	Ensure hfs kernel module is not available	☐	☐
1.1.1.4	Ensure hfsplus kernel module is not available	☐	☐
1.1.1.5	Ensure jffs2 kernel module is not available	☐	☐
1.1.1.6	Ensure unused filesystems kernel modules are not available	☐	☐
1.1.2.1.1	Ensure /tmp is a separate partition	☐	☐
1.1.2.1.2	Ensure nodev option set on /tmp partition	☐	☐
1.1.2.1.3	Ensure nosuid option set on /tmp partition	☐	☐
1.1.2.2.1	Ensure /dev/shm is a separate partition	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition	☐	☐
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition	☐	☐
1.2.1.1	Ensure GPG keys are configured	☐	☐
1.2.1.2	Ensure gpgcheck is configured	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated	☐	☐
1.2.1.4	Ensure package manager repositories are configured	☐	☐
1.3.1	Ensure address space layout randomization is enabled	☐	☐
1.3.2	Ensure ptrace_scope is restricted	☐	☐
1.4.3	Ensure access to /etc/motd is configured	☐	☐
1.4.4	Ensure access to /etc/issue is configured	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured	☐	☐
2.1.1	Ensure time synchronization is in use	☐	☐
2.1.2	Ensure chrony is configured	☐	☐
2.2.1	Ensure xinetd is not installed	☐	☐
2.2.2	Ensure xorg-x11-server-common is not installed	☐	☐
2.2.3	Ensure avahi is not installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
2.2.4	Ensure a print server is not installed	☐	☐
2.2.5	Ensure a dhcp server is not installed	☐	☐
2.2.6	Ensure a dns server is not installed	☐	☐
2.2.7	Ensure FTP client is not installed	☐	☐
2.2.8	Ensure an ftp server is not installed	☐	☐
2.2.9	Ensure a tftp server is not installed	☐	☐
2.2.10	Ensure a web server is not installed	☐	☐
2.2.11	Ensure IMAP and POP3 server is not installed	☐	☐
2.2.12	Ensure Samba is not installed	☐	☐
2.2.13	Ensure HTTP Proxy Server is not installed	☐	☐
2.2.14	Ensure net-snmp is not installed or the snmpd service is not enabled	☐	☐
2.2.15	Ensure NIS server is not installed	☐	☐
2.2.16	Ensure telnet-server is not installed	☐	☐
2.2.17	Ensure mail transfer agent is configured for local-only mode	☐	☐
2.2.18	Ensure nfs-utils is not installed or the nfs-server service is masked	☐	☐
2.2.19	Ensure rsync-daemon is not installed or the rsyncd service is masked	☐	☐
2.3.1	Ensure NIS Client is not installed	☐	☐
2.3.2	Ensure rsh client is not installed	☐	☐
2.3.3	Ensure talk client is not installed	☐	☐
2.3.4	Ensure telnet client is not installed	☐	☐
2.3.5	Ensure LDAP client is not installed	☐	☐
2.3.6	Ensure TFTP client is not installed	☐	☐
3.1.1	Ensure packet redirect sending is disabled	☐	☐
3.1.2	Ensure bogus icmp responses are ignored	☐	☐
3.1.3	Ensure broadcast icmp requests are ignored	☐	☐
3.1.4	Ensure icmp redirects are not accepted	☐	☐
3.1.5	Ensure secure icmp redirects are not accepted	☐	☐
3.1.6	Ensure reverse path filtering is enabled	☐	☐

Recommendation		Set Correctly	
		Yes	No
3.1.7	Ensure source routed packets are not accepted	☐	☐
3.1.8	Ensure suspicious packets are logged	☐	☐
3.1.9	Ensure tcp syn cookies is enabled	☐	☐
3.1.10	Ensure ipv6 router advertisements are not accepted	☐	☐
4.1.1	Ensure iptables is installed	☐	☐
4.1.2	Ensure nftables is not in use	☐	☐
4.1.3	Ensure firewalld is not in use	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured	☐	☐
5.1.7	Ensure permissions on /etc/cron.d are configured	☐	☐
5.1.8	Ensure cron is restricted to authorized users	☐	☐
5.1.9	Ensure at is restricted to authorized users	☐	☐
5.2.1	Ensure access to /etc/ssh/sshd_config is configured	☐	☐
5.2.2	Ensure access to SSH private host key files is configured	☐	☐
5.2.3	Ensure access to SSH public host key files is configured	☐	☐
5.2.4	Ensure sshd Ciphers are configured	☐	☐
5.2.5	Ensure sshd KexAlgorithms is configured	☐	☐
5.2.6	Ensure sshd MACs are configured	☐	☐
5.2.7	Ensure sshd access is configured	☐	☐
5.2.10	Ensure sshd HostbasedAuthentication is disabled	☐	☐
5.2.11	Ensure sshd IgnoreRhosts is enabled	☐	☐
5.2.13	Ensure sshd LogLevel is configured	☐	☐
5.2.14	Ensure sshd MaxAuthTries is configured	☐	☐
5.2.17	Ensure sshd PermitEmptyPasswords is disabled	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled	☐	☐
5.2.20	Ensure sshd UsePAM is enabled	☐	☐
5.3.1	Ensure sudo is installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly	☐	☐
5.4.1	Ensure password creation requirements are configured	☐	☐
5.4.2	Ensure lockout for failed password attempts is configured	☐	☐
5.4.3	Ensure password hashing algorithm is SHA-512	☐	☐
5.4.4	Ensure password reuse is limited	☐	☐
5.5.1.1	Ensure password expiration is 365 days or less	☐	☐
5.5.1.2	Ensure minimum days between password changes is configured	☐	☐
5.5.1.3	Ensure password expiration warning days is 7 or more	☐	☐
5.5.1.4	Ensure inactive password lock is 30 days or less	☐	☐
5.5.1.5	Ensure all users last password change date is in the past	☐	☐
5.5.2	Ensure system accounts are secured	☐	☐
5.5.3	Ensure default group for the root account is GID 0	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive	☐	☐
6.1.1.1.1	Ensure journald service is active	☐	☐
6.1.1.1.2	Ensure journald log file access is configured	☐	☐
6.1.1.1.3	Ensure journald ForwardToSyslog is configured	☐	☐
6.1.1.1.4	Ensure systemd-journal-remote service is not in use	☐	☐
6.1.1.1.5	Ensure journald Storage is configured	☐	☐
6.1.1.1.6	Ensure journald Compress is configured	☐	☐
6.1.2.1	Ensure rsyslog service is enabled and active	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured	☐	☐
6.1.2.3	Ensure rsyslog is not configured to receive logs from a remote client	☐	☐
6.1.3.1	Ensure access to all logfiles has been configured	☐	☐
6.2	Ensure logrotate is configured	☐	☐
7.1.1	Ensure access to /etc/passwd is configured	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured	☐	☐
7.1.3	Ensure access to /etc/group is configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
7.1.4	Ensure access to /etc/group- is configured	☐	☐
7.1.5	Ensure access to /etc/shadow is configured	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured	☐	☐
7.1.7	Ensure access to /etc/gshadow is configured	☐	☐
7.1.8	Ensure access to /etc/gshadow- is configured	☐	☐
7.1.9	Ensure access to /etc/shells is configured	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured	☐	☐
7.1.11	Ensure world writable files and directories are secured	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist	☐	☐
7.2.1	Ensure accounts in /etc/passwd use shadowed passwords	☐	☐
7.2.2	Ensure /etc/shadow password fields are not empty	☐	☐
7.2.3	Ensure all groups in /etc/passwd exist in /etc/group	☐	☐
7.2.8	Ensure local interactive user home directories are configured	☐	☐
7.2.9	Ensure local interactive user dot files access is configured	☐	☐

Appendix: CIS Controls v8 IG 3 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1.1	Ensure cramfs kernel module is not available	☐	☐
1.1.1.2	Ensure freevxfs kernel module is not available	☐	☐
1.1.1.3	Ensure hfs kernel module is not available	☐	☐
1.1.1.4	Ensure hfsplus kernel module is not available	☐	☐
1.1.1.5	Ensure jffs2 kernel module is not available	☐	☐
1.1.1.6	Ensure unused filesystems kernel modules are not available	☐	☐
1.1.2.1.1	Ensure /tmp is a separate partition	☐	☐
1.1.2.1.2	Ensure nodev option set on /tmp partition	☐	☐
1.1.2.1.3	Ensure nosuid option set on /tmp partition	☐	☐
1.1.2.2.1	Ensure /dev/shm is a separate partition	☐	☐
1.1.2.2.2	Ensure nodev option set on /dev/shm partition	☐	☐
1.1.2.2.3	Ensure nosuid option set on /dev/shm partition	☐	☐
1.2.1.1	Ensure GPG keys are configured	☐	☐
1.2.1.2	Ensure gpgcheck is configured	☐	☐
1.2.1.3	Ensure TDNF gpgcheck is globally activated	☐	☐
1.2.1.4	Ensure package manager repositories are configured	☐	☐
1.3.1	Ensure address space layout randomization is enabled	☐	☐
1.3.2	Ensure ptrace_scope is restricted	☐	☐
1.4.3	Ensure access to /etc/motd is configured	☐	☐
1.4.4	Ensure access to /etc/issue is configured	☐	☐
1.4.5	Ensure access to /etc/issue.net is configured	☐	☐
2.1.1	Ensure time synchronization is in use	☐	☐
2.1.2	Ensure chrony is configured	☐	☐
2.2.1	Ensure xinetd is not installed	☐	☐
2.2.2	Ensure xorg-x11-server-common is not installed	☐	☐
2.2.3	Ensure avahi is not installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
2.2.4	Ensure a print server is not installed	☐	☐
2.2.5	Ensure a dhcp server is not installed	☐	☐
2.2.6	Ensure a dns server is not installed	☐	☐
2.2.7	Ensure FTP client is not installed	☐	☐
2.2.8	Ensure an ftp server is not installed	☐	☐
2.2.9	Ensure a tftp server is not installed	☐	☐
2.2.10	Ensure a web server is not installed	☐	☐
2.2.11	Ensure IMAP and POP3 server is not installed	☐	☐
2.2.12	Ensure Samba is not installed	☐	☐
2.2.13	Ensure HTTP Proxy Server is not installed	☐	☐
2.2.14	Ensure net-snmp is not installed or the snmpd service is not enabled	☐	☐
2.2.15	Ensure NIS server is not installed	☐	☐
2.2.16	Ensure telnet-server is not installed	☐	☐
2.2.17	Ensure mail transfer agent is configured for local-only mode	☐	☐
2.2.18	Ensure nfs-utils is not installed or the nfs-server service is masked	☐	☐
2.2.19	Ensure rsync-daemon is not installed or the rsyncd service is masked	☐	☐
2.3.1	Ensure NIS Client is not installed	☐	☐
2.3.2	Ensure rsh client is not installed	☐	☐
2.3.3	Ensure talk client is not installed	☐	☐
2.3.4	Ensure telnet client is not installed	☐	☐
2.3.5	Ensure LDAP client is not installed	☐	☐
2.3.6	Ensure TFTP client is not installed	☐	☐
3.1.1	Ensure packet redirect sending is disabled	☐	☐
3.1.2	Ensure bogus icmp responses are ignored	☐	☐
3.1.3	Ensure broadcast icmp requests are ignored	☐	☐
3.1.4	Ensure icmp redirects are not accepted	☐	☐
3.1.5	Ensure secure icmp redirects are not accepted	☐	☐
3.1.6	Ensure reverse path filtering is enabled	☐	☐

Recommendation		Set Correctly	
		Yes	No
3.1.7	Ensure source routed packets are not accepted	☐	☐
3.1.8	Ensure suspicious packets are logged	☐	☐
3.1.9	Ensure tcp syn cookies is enabled	☐	☐
3.1.10	Ensure ipv6 router advertisements are not accepted	☐	☐
4.1.1	Ensure iptables is installed	☐	☐
4.1.2	Ensure nftables is not in use	☐	☐
4.1.3	Ensure firewalld is not in use	☐	☐
5.1.2	Ensure permissions on /etc/crontab are configured	☐	☐
5.1.3	Ensure permissions on /etc/cron.hourly are configured	☐	☐
5.1.4	Ensure permissions on /etc/cron.daily are configured	☐	☐
5.1.5	Ensure permissions on /etc/cron.weekly are configured	☐	☐
5.1.6	Ensure permissions on /etc/cron.monthly are configured	☐	☐
5.1.7	Ensure permissions on /etc/cron.d are configured	☐	☐
5.1.8	Ensure cron is restricted to authorized users	☐	☐
5.1.9	Ensure at is restricted to authorized users	☐	☐
5.2.1	Ensure access to /etc/ssh/sshd_config is configured	☐	☐
5.2.2	Ensure access to SSH private host key files is configured	☐	☐
5.2.3	Ensure access to SSH public host key files is configured	☐	☐
5.2.4	Ensure sshd Ciphers are configured	☐	☐
5.2.5	Ensure sshd KexAlgorithms is configured	☐	☐
5.2.6	Ensure sshd MACs are configured	☐	☐
5.2.7	Ensure sshd access is configured	☐	☐
5.2.10	Ensure sshd HostbasedAuthentication is disabled	☐	☐
5.2.11	Ensure sshd IgnoreRhosts is enabled	☐	☐
5.2.13	Ensure sshd LogLevel is configured	☐	☐
5.2.14	Ensure sshd MaxAuthTries is configured	☐	☐
5.2.17	Ensure sshd PermitEmptyPasswords is disabled	☐	☐
5.2.18	Ensure sshd PermitRootLogin is disabled	☐	☐
5.2.20	Ensure sshd UsePAM is enabled	☐	☐
5.3.1	Ensure sudo is installed	☐	☐

Recommendation		Set Correctly	
		Yes	No
5.3.2	Ensure re-authentication for privilege escalation is not disabled globally	☐	☐
5.3.3	Ensure sudo authentication timeout is configured correctly	☐	☐
5.4.1	Ensure password creation requirements are configured	☐	☐
5.4.2	Ensure lockout for failed password attempts is configured	☐	☐
5.4.3	Ensure password hashing algorithm is SHA-512	☐	☐
5.4.4	Ensure password reuse is limited	☐	☐
5.5.1.1	Ensure password expiration is 365 days or less	☐	☐
5.5.1.2	Ensure minimum days between password changes is configured	☐	☐
5.5.1.3	Ensure password expiration warning days is 7 or more	☐	☐
5.5.1.4	Ensure inactive password lock is 30 days or less	☐	☐
5.5.1.5	Ensure all users last password change date is in the past	☐	☐
5.5.2	Ensure system accounts are secured	☐	☐
5.5.3	Ensure default group for the root account is GID 0	☐	☐
5.5.4	Ensure default user umask is 027 or more restrictive	☐	☐
6.1.1.1.1	Ensure journald service is active	☐	☐
6.1.1.1.2	Ensure journald log file access is configured	☐	☐
6.1.1.1.3	Ensure journald ForwardToSyslog is configured	☐	☐
6.1.1.1.4	Ensure systemd-journal-remote service is not in use	☐	☐
6.1.1.1.5	Ensure journald Storage is configured	☐	☐
6.1.1.1.6	Ensure journald Compress is configured	☐	☐
6.1.2.1	Ensure rsyslog service is enabled and active	☐	☐
6.1.2.2	Ensure rsyslog log file creation mode is configured	☐	☐
6.1.2.3	Ensure rsyslog is not configured to receive logs from a remote client	☐	☐
6.1.3.1	Ensure access to all logfiles has been configured	☐	☐
6.2	Ensure logrotate is configured	☐	☐
7.1.1	Ensure access to /etc/passwd is configured	☐	☐
7.1.2	Ensure access to /etc/passwd- is configured	☐	☐
7.1.3	Ensure access to /etc/group is configured	☐	☐

Recommendation		Set Correctly	
		Yes	No
7.1.4	Ensure access to /etc/group- is configured	☐	☐
7.1.5	Ensure access to /etc/shadow is configured	☐	☐
7.1.6	Ensure access to /etc/shadow- is configured	☐	☐
7.1.7	Ensure access to /etc/gshadow is configured	☐	☐
7.1.8	Ensure access to /etc/gshadow- is configured	☐	☐
7.1.9	Ensure access to /etc/shells is configured	☐	☐
7.1.10	Ensure access to /etc/security/opasswd is configured	☐	☐
7.1.11	Ensure world writable files and directories are secured	☐	☐
7.1.12	Ensure no files or directories without an owner and a group exist	☐	☐
7.2.1	Ensure accounts in /etc/passwd use shadowed passwords	☐	☐
7.2.2	Ensure /etc/shadow password fields are not empty	☐	☐
7.2.3	Ensure all groups in /etc/passwd exist in /etc/group	☐	☐
7.2.8	Ensure local interactive user home directories are configured	☐	☐
7.2.9	Ensure local interactive user dot files access is configured	☐	☐

Appendix: CIS Controls v8 Unmapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.3.3	Ensure core dump backtraces are disabled	☐	☐
1.3.4	Ensure core dump storage is disabled	☐	☐
1.4.1	Ensure local login warning banner is configured properly	☐	☐
1.4.2	Ensure remote login warning banner is configured properly	☐	☐
5.1.1	Ensure cron daemon is enabled	☐	☐

Appendix: Change History

Date	Version	Changes for this version
08/01/2025	1.0.0	PUBLISHED