Ubuntu 22.04 Meta STIG Rules

V-260469

Title

Ubuntu 22.04 LTS must disable the x86 Ctrl-Alt-Delete key sequence.

Description

A locally logged-on user who presses Ctrl-Alt-Delete, when at the console, can reboot the system. If accidentally pressed, as could happen in the case of a mixed OS environment, this can create the risk of short-term loss of availability of systems due to unintentional reboot.

Rule Check

Verify Ubuntu 22.04 LTS is not configured to reboot the system when Ctrl-Alt-Delete is pressed by using the following command:

```
$ systemctl status ctrl-alt-del.target
ctrl-alt-del.target
     Loaded: masked (Reason: Unit ctrl-alt-del.target is masked.)
     Active: inactive (dead)
```

If the "ctrl-alt-del.target" is not masked, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to disable the Ctrl-Alt-Delete sequence for the command line by using the following commands:

```
$ sudo systemctl disable ctrl-alt-del.target

$ sudo systemctl mask ctrl-alt-del.target
```

Reload the daemon to take effect:

```
$ sudo systemctl daemon-reload
```

V-260470

Title

Ubuntu 22.04 LTS, when booted, must require authentication upon booting into single-user and maintenance modes.

Description

To mitigate the risk of unauthorized access to sensitive information by entities that have been issued certificates by DOD-approved PKIs, all DOD systems (e.g., web servers and web portals) must be properly configured to incorporate access control methods that do not rely solely on the possession of a certificate for access.

Successful authentication must not automatically give an entity access to an asset or security boundary. Authorization procedures and controls must be implemented to ensure each authenticated entity also has a validated and current authorization. Authorization is the process of determining whether an entity, once authenticated, is permitted to access a specific asset. Information systems use access control policies and enforcement mechanisms to implement this requirement.

Access control policies include identity-based policies, role-based policies, and attribute-based policies. Access enforcement mechanisms include access control lists, access control matrices, and cryptography. These policies and mechanisms must be employed by the application to control access between users (or processes acting on behalf of users) and objects (e.g., devices, files, records, processes, programs, and domains) in the information system.

Rule Check

Verify Ubuntu 22.04 LTS requires a password for authentication upon booting into single-user and maintenance modes by using the following command:

```
$ sudo grep -i password /boot/grub/grub.cfg

password_pbkdf2 root grub.pbkdf2.sha512.10000.03255F190F0E2F7B4F0D1C3216012309162F022A7A636771
```

If the root password entry does not begin with "password_pbkdf2", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to require a password for authentication upon booting into single-user and maintenance modes.

Generate an encrypted (grub) password for root by using the following command:

```
$ grub-mkpasswd-pbkdf2
Enter Password:
Reenter Password:
PBKDF2 hash of your password is grub.pbkdf2.sha512.10000.03255F190F0E2F7B4F0D1C3216012309162F022A7A636771
```

Using the hash from the output, modify the "/etc/grub.d/40_custom" file by using the following command to add a boot password:

```
$ sudo sed -i '$i set superusers=\"root\"\npassword_pbkdf2 root <hash>' /etc/grub.d/40_custom
```

where is the hash generated by grub-mkpasswd-pbkdf2 command.

Generate an updated "grub.conf" file with the new password by using the following command:

```
$ sudo update-grub
```

V-260471

Title

Ubuntu 22.04 LTS must initiate session audits at system startup.

Description

If auditing is enabled late in the startup process, the actions of some startup processes may not be audited. Some audit systems also maintain state information only available if auditing is enabled before a given process is created.

Rule Check

Verify that Ubuntu 22.04 LTS enables auditing at system startup in grub by using the following command:

```
$ grep "^\s*linux" /boot/grub/grub.cfg

linux   /vmlinuz-5.15.0-89-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro audit=1
     linux   /vmlinuz-5.15.0-89-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro audit=1
     linux   /vmlinuz-5.15.0-89-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro single nomodeset dis_ucode_ldr audit=1
     linux   /vmlinuz-5.15.0-83-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro audit=1
     linux   /vmlinuz-5.15.0-83-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro single nomodeset dis_ucode_ldr audit=1
```

If any linux lines do not contain "audit=1", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to produce audit records at system startup.

Edit the "/etc/default/grub" file and add "audit=1" to the "GRUB_CMDLINE_LINUX" option.

To update the grub config file, run:

```
$ sudo update-grub
```

V-260472

Title

Ubuntu 22.04 LTS must restrict access to the kernel message buffer.

Description

Restricting access to the kernel message buffer limits access only to root. This prevents attackers from gaining additional system information as a nonprivileged user.

Rule Check

Verify Ubuntu 22.04 LTS is configured to restrict access to the kernel message buffer by using the following command:

```
$ sysctl kernel.dmesg_restrict
kernel.dmesg_restrict = 1
```

If "kernel.dmesg_restrict" is not set to "1" or is missing, this is a finding.

Verify that there are no configurations that enable the kernel dmesg function:

```
$ sudo grep -ir kernel.dmesg_restrict /run/sysctl.d/* /etc/sysctl.d/* /usr/local/lib/sysctl.d/* /usr/lib/sysctl.d/* /lib/sysctl.d/* /etc/sysctl.conf 2> /dev/null
/etc/sysctl.d/10-kernel-hardening.conf:kernel.dmesg_restrict = 1
```

If "kernel.dmesg_restrict" is not set to "1", is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to restrict access to the kernel message buffer.

Add or modify the following line in the "/etc/sysctl.conf" file:

kernel.dmesg_restrict = 1

Remove any configurations that conflict with the above from the following locations:

/run/sysctl.d/ /etc/sysctl.d/ /usr/local/lib/sysctl.d/ /usr/lib/sysctl.d/ /lib/sysctl.d/ /etc/sysctl.conf

Reload settings from all system configuration files by using the following command:

```
$ sudo sysctl --system
```

V-260473

Title

Ubuntu 22.04 LTS must disable kernel core dumps so that it can fail to a secure state if system initialization fails, shutdown fails or aborts fail.

Description

Kernel core dumps may contain the full contents of system memory at the time of the crash. Kernel core dumps may consume a considerable amount of disk space and may result in denial of service by exhausting the available space on the target file system partition.

Rule Check

Verify that kernel core dumps are disabled unless needed by using the following command:

```
$ systemctl status kdump.service
kdump.service
     Loaded: masked (Reason: Unit kdump.service is masked.)
     Active: inactive (dead)
```

If "kdump.service" is not masked and inactive, ask the system administrator (SA) if the use of the service is required and documented with the information system security officer (ISSO).

If the service is active and is not documented, this is a finding.

Fix

If kernel core dumps are not required, disable and mask "kdump.service" by using the following command:

```
$ sudo systemctl mask kdump --now
```

If kernel core dumps are required, document the need with the ISSO.

V-260474

Title

Ubuntu 22.04 LTS must implement address space layout randomization to protect its memory from unauthorized code execution.

Description

Some adversaries launch attacks with the intent of executing code in nonexecutable regions of memory or in prohibited memory locations. Security safeguards employed to protect memory include, for example, data execution prevention and address space layout randomization. Data execution prevention safeguards can either be hardware-enforced or software-enforced with hardware providing the greater strength of mechanism.

Examples of attacks are buffer overflow attacks.

Rule Check

Verify Ubuntu 22.04 LTS implements address space layout randomization (ASLR) by using the following command:

```
$ sysctl kernel.randomize_va_space
kernel.randomize_va_space = 2
```

If no output is returned, verify the kernel parameter "randomize_va_space" is set to "2" by using the following command:

```
$ cat /proc/sys/kernel/randomize_va_space
2
```

If "kernel.randomize_va_space" is not set to "2", this is a finding.

Verify that a saved value of the "kernel.randomize_va_space" variable is not defined.

```
$ sudo grep -ER "^kernel.randomize_va_space=[^2]" /etc/sysctl.conf /etc/sysctl.d
```

If this returns a result, this is a finding.

Fix

Remove the "kernel.randomize_va_space" entry found in the "/etc/sysctl.conf" file or any file located in the "/etc/sysctl.d/" directory.

Reload the system configuration files for the changes to take effect by using the following command:

```
$ sudo sysctl --system
```

V-260475

Title

Ubuntu 22.04 LTS must implement nonexecutable data to protect its memory from unauthorized code execution.

Description

Some adversaries launch attacks with the intent of executing code in nonexecutable regions of memory or in memory locations that are prohibited. Security safeguards employed to protect memory include, for example, data execution prevention and address space layout randomization. Data execution prevention safeguards can either be hardware-enforced or software-enforced with hardware providing the greater strength of mechanism.

Examples of attacks are buffer overflow attacks.

Rule Check

Verify the NX (no-execution) bit flag is set on the system by using the following command:

```
$ sudo dmesg | grep -i "execute disable"
[    0.000000] NX (Execute Disable) protection: active
```

If "dmesg" does not show "NX (Execute Disable) protection: active", check the hardware capabilities of the installed CPU by using the following command:

```
$ grep flags /proc/cpuinfo | grep -o nx | sort -u
nx
```

If no output is returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enable NX.

If the installed CPU is hardware capable of NX protection, check if the system's BIOS/UEFI setup configuration permits toggling the "NX bit" or "no execution bit", and set it to "enabled".

V-260476

Title

Ubuntu 22.04 LTS must be configured so that the Advance Package Tool (APT) prevents the installation of patches, service packs, device drivers, or operating system components without verification they have been digitally signed using a certificate that is recognized and approved by the organization.

Description

Changes to any software components can have significant effects on the overall security of the operating system. This requirement ensures the software has not been tampered with and that it has been provided by a trusted vendor.

Accordingly, patches, service packs, device drivers, or operating system components must be signed with a certificate recognized and approved by the organization.

Verifying the authenticity of the software prior to installation validates the integrity of the patch or upgrade received from a vendor. This ensures the software has not been tampered with and that it has been provided by a trusted vendor. Self-signed certificates are disallowed by this requirement. The operating system should not have to verify the software again. This requirement does not mandate DOD certificates for this purpose; however, the certificate used to verify the software must be from an approved certificate authority (CA).

Rule Check

Verify that APT is configured to prevent the installation of patches, service packs, device drivers, or Ubuntu operating system components without verification they have been digitally signed using a certificate that is recognized and approved by the organization by using the following command:

```
$ grep -i allowunauthenticated /etc/apt/apt.conf.d/*
/etc/apt/apt.conf.d/01-vendor-ubuntu:APT::Get::AllowUnauthenticated "false";
```

If "APT::Get::AllowUnauthenticated" is not set to "false", is commented out, or is missing, this is a finding.

Fix

Configure APT to prevent the installation of patches, service packs, device drivers, or Ubuntu operating system components without verification they have been digitally signed using a certificate that is recognized and approved by the organization.

Add or modify the following line in any file under the "/etc/apt/apt.conf.d/" directory:

APT::Get::AllowUnauthenticated "false";

V-260477

Title

Ubuntu 22.04 LTS must be configured so that the Advance Package Tool (APT) removes all software components after updated versions have been installed.

Description

Previous versions of software components that are not removed from the information system after updates have been installed may be exploited by adversaries. Some information technology products may remove older versions of software automatically from the information system.

Rule Check

Verify APT is configured to remove all software components after updated versions have been installed by using the following command:

```
$ grep -i remove-unused /etc/apt/apt.conf.d/50-unattended-upgrades
Unattended-Upgrade::Remove-Unused-Kernel-Packages "true";
Unattended-Upgrade::Remove-Unused-Dependencies "true";
```

If "Unattended-Upgrade::Remove-Unused-Kernel-Packages" and "Unattended-Upgrade::Remove-Unused-Dependencies" are not set to "true", are commented out, or are missing, this is a finding.

Fix

Configure APT to remove all software components after updated versions have been installed.

Add or modify the following lines in the "/etc/apt/apt.conf.d/50-unattended-upgrades" file:

Unattended-Upgrade::Remove-Unused-Kernel-Packages "true"; Unattended-Upgrade::Remove-Unused-Dependencies "true";

V-260478

Title

Ubuntu 22.04 LTS must have the "libpam-pwquality" package installed.

Description

Use of a complex password helps to increase the time and resources required to compromise the password. Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks. "pwquality" enforces complex password construction configuration and has the ability to limit brute-force attacks on the system.

Rule Check

Verify Ubuntu 22.04 LTS has the "libpam-pwquality" package installed with the following command:

```
$ dpkg -l | grep libpam-pwquality
ii     libpam-pwquality:amd64     1.4.4-1build2     amd64     PAM module to check password strength
```

If "libpam-pwquality" is not installed, this is a finding.

Fix

Install the "pam_pwquality" package by using the following command:

```
$ sudo apt-get install libpam-pwquality
```

V-260479

Title

Ubuntu 22.04 LTS must have the "chrony" package installed.

Description

Inaccurate time stamps make it more difficult to correlate events and can lead to an inaccurate analysis. Determining the correct time a particular event occurred on a system is critical when conducting forensic analysis and investigating system events. Sources outside the configured acceptable allowance (drift) may be inaccurate.

Organizations must consider endpoints that may not have regular access to the authoritative time server (e.g., mobile, teleworking, and tactical endpoints).

Rule Check

Verify the "chrony" package is installed using the following command:

```
$ dpkg -l | grep chrony
ii     chrony     4.2-2ubuntu2     amd64     Versatile implementation of the Network Time Protocol
```

If the "chrony" package is not installed,Â this is a finding.

Fix

Install the "chrony" network time protocol package using the following command:

```
$ sudo apt-get install chrony
```

V-260480

Title

Ubuntu 22.04 LTS must not have the "systemd-timesyncd" package installed.

Description

Inaccurate time stamps make it more difficult to correlate events and can lead to an inaccurate analysis. Determining the correct time a particular event occurred on a system is critical when conducting forensic analysis and investigating system events. Sources outside the configured acceptable allowance (drift) may be inaccurate.

Organizations must consider endpoints that may not have regular access to the authoritative time server (e.g., mobile, teleworking, and tactical endpoints).

Rule Check

Verify that the "systemd-timesyncd" package is not installed by using the following command:

```
$ dpkg -l | grep systemd-timesyncd
```

If the "systemd-timesyncd" package is installed, this is a finding.

Fix

The "systemd-timesyncd" package will be uninstalled as part of the "chrony" package install. The remaining configuration files for "systemd-timesyncd" must be purged from the operating system:

```
$ sudo dpkg -P --force-all systemd-timesyncd
```

V-260481

Title

Ubuntu 22.04 LTS must not have the "ntp" package installed.

Description

Inaccurate time stamps make it more difficult to correlate events and can lead to an inaccurate analysis. Determining the correct time a particular event occurred on a system is critical when conducting forensic analysis and investigating system events. Sources outside the configured acceptable allowance (drift) may be inaccurate.

Organizations must consider endpoints that may not have regular access to the authoritative time server (e.g., mobile, teleworking, and tactical endpoints).

Rule Check

Verify that the "ntp" package is not installed by using the following command:

```
$ dpkg -l | grep ntp
```

If the "ntp" package is installed, this is a finding.

Fix

Uninstall the "ntp" package by using the following command:

```
$ sudo dpkg -P --force-all ntp
```

V-260482

Title

Ubuntu 22.04 LTS must not have the "rsh-server" package installed.

Description

It is detrimental for operating systems to provide, or install by default, functionality exceeding requirements or mission objectives. These unnecessary capabilities or services are often overlooked and therefore may remain unsecured. They increase the risk to the platform by providing additional attack vectors.

Remote Shell (RSH) is a client/server application protocol that provides an unencrypted remote access service, which does not provide for the confidentiality and integrity of user passwords or the remote session. If users were allowed to login to a system using RSH, the privileged user passwords and communications could be compromised.

Removing the "rsh-server" package decreases the risk of accidental or intentional activation of the RSH service.

Rule Check

Verify the "rsh-server" package is not installed by using the following command:

```
$ dpkg -l | grep rsh-server
```

If the "rsh-server" package is installed, this is a finding.

Fix

Remove the "rsh-server" package by using the following command:

```
$ sudo apt-get remove rsh-server
```

V-260483

Title

Ubuntu 22.04 LTS must not have the "telnet" package installed.

Description

It is detrimental for operating systems to provide, or install by default, functionality exceeding requirements or mission objectives. These unnecessary capabilities are often overlooked and therefore, may remain unsecure. They increase the risk to the platform by providing additional attack vectors.

Telnet is a client/server application protocol that provides an unencrypted remote access service, which does not provide for the confidentiality and integrity of user passwords or the remote session. If users were allowed to login to a system using Telnet, the privileged user passwords and communications could be compromised.

Removing the "telnetd" package decreases the risk of accidental or intentional activation of the Telnet service.

Rule Check

Verify that the "telnetd" package is not installed on Ubuntu 22.04 LTS by using the following command:

```
$ dpkg -l | grep telnetd
```

If the "telnetd" package is installed, this is a finding.

Fix

Remove the "telnetd" package by using the following command:

```
$ sudo apt-get remove telnetd
```

V-260484

Title

Ubuntu 22.04 LTS must implement cryptographic mechanisms to prevent unauthorized disclosure and modification of all information that requires protection at rest.

Description

Operating systems handling data requiring "data at rest" protections must employ cryptographic mechanisms to prevent unauthorized disclosure and modification of the information at rest.

Selection of a cryptographic mechanism is based on the need to protect the integrity of organizational information. The strength of the mechanism is commensurate with the security category and/or classification of the information. Organizations have the flexibility to either encrypt all information on storage devices (i.e., full disk encryption) or encrypt specific data structures (e.g., files, records, or fields).

Satisfies: SRG-OS-000185-GPOS-00079, SRG-OS-000404-GPOS-00183, SRG-OS-000405-GPOS-00184

Rule Check

Verify Ubuntu 22.04 LTS prevents unauthorized disclosure or modification of all information requiring at-rest protection by using disk encryption.

Note: If there is a documented and approved reason for not having data-at-rest encryption, this requirement is not applicable.

Determine the partition layout for the system by using the following command:

```
$ sudo fdisk -l

...
Device               Start               End        Sectors       Size  Type
/dev/sda1         2048      2203647      2201600          1G  EFI System
/dev/sda2  2203648      6397951       4194304          2G  Linux filesystem
/dev/sda3  6397952  536868863  530470912  252.9G  Linux filesystem
...
```

Verify the system partitions are all encrypted by using the following command:

```
# more /etc/crypttab
```

Every persistent disk partition present must have an entry in the file.

If any partitions other than the boot partition or pseudo file systems (such as /proc or /sys) are not listed, this is a finding.

Fix

To encrypt an entire partition, dedicate a partition for encryption in the partition layout.

Note: Encrypting a partition in an already-installed system is more difficult because it will need to be resized and existing partitions changed.

V-260485

Title

Ubuntu 22.04 LTS must have directories that contain system commands set to a mode of "755" or less permissive.

Description

Protecting audit information also includes identifying and protecting the tools used to view and manipulate log data. Therefore, protecting audit tools is necessary to prevent unauthorized operation on audit information.

Operating systems providing tools to interface with audit information will leverage user permissions and roles identifying the user accessing the tools and the corresponding rights the user has in order to make access decisions regarding the deletion of audit tools.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

Rule Check

Verify the system commands directories have mode "755" or less permissive by using the following command:

```
$ find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin -perm /022 -type d -exec stat -c "%n %a" '{}' \;
```

If any directories are found to be group-writable or world-writable, this is a finding.

Fix

Configure Ubuntu 22.04 LTS commands directories to be protected from unauthorized access. Run the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin -perm /022 -type d -exec chmod -R 755 '{}' \;
```

V-260486

Title

Ubuntu 22.04 LTS must have system commands set to a mode of "755" or less permissive.

Description

If Ubuntu 22.04 LTS were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to Ubuntu 22.04 LTS with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the system commands contained in the following directories have mode "755" or less permissive by using the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin -perm /022 -type f -exec stat -c "%n %a" '{}' \;
```

If any files are found to be group-writable or world-writable, this is a finding.

Fix

Configure Ubuntu 22.04 LTS commands to be protected from unauthorized access. Run the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin -perm /022 -type f -exec chmod 755 '{}' \;
```

V-260487

Title

Ubuntu 22.04 LTS library files must have mode "755" or less permissive.

Description

If the operating system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to operating systems with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the systemwide shared library files contained in the directories "/lib", "/lib64", and "/usr/lib" have mode "755" or less permissive by using the following command:

```
$ sudo find /lib /lib64 /usr/lib -perm /022 -type f -exec stat -c "%n %a" '{}' \;
```

If any files are found to be group-writable or world-writable, this is a finding.

Fix

Configure the library files to be protected from unauthorized access. Run the following command:

```
$ sudo find /lib /lib64 /usr/lib -perm /022 -type f -exec chmod 755 '{}' \;
```

V-260488

Title

Ubuntu 22.04 LTS must configure the "/var/log" directory to have mode "755" or less permissive.

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify the "/var/log" directory has mode of "755" or less permissive by using the following command:

Note: If rsyslog is active and enabled on the operating system, this requirement is not applicable.

```
$ stat -c "%n %a" /var/log
/var/log 755
```

If a value of "755" or less permissive is not returned, this is a finding.

Fix

Configure the "/var/log" directory to have permissions of "0755" by using the following command:

```
$ sudo chmod 0755 /var/log
```

V-260489

Title

Ubuntu 22.04 LTS must generate error messages that provide information necessary for corrective actions without revealing information that could be exploited by adversaries.

Description

Any operating system providing too much information in error messages risks compromising the data and security of the structure, and content of error messages needs to be carefully considered by the organization.

Organizations carefully consider the structure/content of error messages. The extent to which information systems are able to identify and handle error conditions is guided by organizational policy and operational requirements. Information that could be exploited by adversaries includes, for example, erroneous logon attempts with passwords entered by mistake as the username, mission/business information that can be derived from (if not stated explicitly by) information recorded, and personal information, such as account numbers, social security numbers, and credit card numbers.

The /var/log/btmp, /var/log/wtmp, and /var/log/lastlog files have group write and global read permissions to allow for the lastlog function to perform. Limiting the permissions beyond this configuration will result in the failure of functions that rely on the lastlog database.

Rule Check

Verify Ubuntu 22.04 LTS has all system log files under the "/var/log" directory with a permission set to "640" or less permissive by using the following command:

Note: The btmp, wtmp, and lastlog files are excluded. Refer to the Discussion for details.

```
$ sudo find /var/log -perm /137 ! -name '*[bw]tmp' ! -name '*lastlog' -type f -exec stat -c "%n %a" {} \;
```

If the command displays any output, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to set permissions of all log files under the "/var/log" directory to "640" or more restricted by using the following command:

Note: The btmp, wtmp, and lastlog files are excluded. Refer to the Discussion for details.

```
$ sudo find /var/log -perm /137 ! -name '*[bw]tmp' ! -name '*lastlog' -type f -exec chmod 640 '{}' \;
```

V-260490

Title

Ubuntu 22.04 LTS must generate system journal entries without revealing information that could be exploited by adversaries.

Description

Any operating system providing too much information in error messages risks compromising the data and security of the structure, and content of error messages needs to be carefully considered by the organization.

Organizations carefully consider the structure/content of error messages. The extent to which information systems are able to identify and handle error conditions is guided by organizational policy and operational requirements. Information that could be exploited by adversaries includes, for example, erroneous logon attempts with passwords entered by mistake as the username, mission/business information that can be derived from (if not stated explicitly by) information recorded, and personal information, such as account numbers, social security numbers, and credit card numbers.

Rule Check

Verify the /run/log/journal and /var/log/journal directories have permissions set to "2640" or less permissive by using the following command:

```
$ sudo find /run/log/journal /var/log/journal  -type d -exec stat -c "%n %a" {} \;
/run/log/journal 2640
/var/log/journal 2640
/var/log/journal/3b018e681c904487b11671b9c1987cce 2640
```

If any output returned has a permission set greater than "2640", this is a finding.

Verify all files in the /run/log/journal and /var/log/journal directories have permissions set to "640" or less permissive by using the following command:

```
$ sudo find /run/log/journal /var/log/journal  -type f -exec stat -c "%n %a" {} \;
/var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000003c7a-0006073f8d1c0fec.journal 640
/var/log/journal/3b018e681c904487b11671b9c1987cce/system.journal 640
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000.journal 640
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000@bdedf14602ff4081a77dc7a6debc8626-00000000000062a6-00060b4b414b617a.journal 640
/var/log/journal/3b018e681c904487b11671b9c1987cce
```

If any output returned has a permission set greater than "640", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to set the appropriate permissions to the files and directories used by the systemd journal:

Add or modify the following lines in the "/usr/lib/tmpfiles.d/systemd.conf" file:

z /run/log/journal 2640 root systemd-journal - - Z /run/log/journal/%m ~2640 root systemd-journal - - z /var/log/journal 2640 root systemd-journal - - z /var/log/journal/%m 2640 root systemd-journal - - z /var/log/journal/%m/system.journal 0640 root systemd-journal - -

Restart the system for the changes to take effect.

V-260491

Title

Ubuntu 22.04 LTS must configure "/var/log/syslog" file with mode "640" or less permissive.

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify that Ubuntu 22.04 LTS configures the "/var/log/syslog" file with mode "640" or less permissive by using the following command:

```
$ stat -c "%n %a" /var/log/syslog
/var/log/syslog 640
```

If a value of "640" or less permissive is not returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to have permissions of "640" for the "/var/log/syslog" file by using the following command:

```
$ sudo chmod 0640 /var/log/syslog
```

V-260492

Title

Ubuntu 22.04 LTS must configure audit tools with a mode of "755" or less permissive.

Description

Protecting audit information also includes identifying and protecting the tools used to view and manipulate log data. Therefore, protecting audit tools is necessary to prevent unauthorized operation on audit information.

Operating systems providing tools to interface with audit information will leverage user permissions and roles identifying the user accessing the tools and the corresponding rights the user enjoys in order to make access decisions regarding the access to audit tools.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

Satisfies: SRG-OS-000256-GPOS-00097, SRG-OS-000257-GPOS-00098

Rule Check

Verify Ubuntu 22.04 LTS configures the audit tools to have a file permission of "755" or less to prevent unauthorized access by using the following command:

```
$ stat -c "%n %a" /sbin/auditctl /sbin/aureport /sbin/ausearch /sbin/autrace /sbin/auditd /sbin/audispd* /sbin/augenrules
/sbin/auditctl 755
/sbin/aureport 755
/sbin/ausearch 755
/sbin/autrace 755
/sbin/auditd 755
/sbin/audispd-zos-remote 755
/sbin/augenrules 755
```

If any of the audit tools have a mode more permissive than "0755", this is a finding.

Fix

Configure the audit tools on Ubuntu 22.04 LTS to be protected from unauthorized access by setting the correct permissive mode using the following command:

```
$ sudo chmod 755 <audit_tool_name>
```

Replace "" with the audit tool that does not have the correct permissions.

V-260493

Title

Ubuntu 22.04 LTS must have directories that contain system commands owned by "root".

Description

Protecting audit information also includes identifying and protecting the tools used to view and manipulate log data. Therefore, protecting audit tools is necessary to prevent unauthorized operation on audit information.

Operating systems providing tools to interface with audit information will leverage user permissions and roles identifying the user accessing the tools and the corresponding rights the user has in order to make access decisions regarding the deletion of audit tools.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

Rule Check

Verify the system commands directories are owned by "root" by using the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin ! -user root -type d -exec stat -c "%n %U" '{}' \;
```

If any system commands directories are returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS commands directories to be protected from unauthorized access. Run the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin ! -user root -type d -exec chown root '{}' \;
```

V-260494

Title

Ubuntu 22.04 LTS must have directories that contain system commands group-owned by "root".

Description

Protecting audit information also includes identifying and protecting the tools used to view and manipulate log data. Therefore, protecting audit tools is necessary to prevent unauthorized operation on audit information.

Operating systems providing tools to interface with audit information will leverage user permissions and roles identifying the user accessing the tools and the corresponding rights the user has in order to make access decisions regarding the deletion of audit tools.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

Rule Check

Verify the system commands directories are group-owned by "root" by using the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin ! -group root -type d -exec stat -c "%n %G" '{}' \;
```

If any system commands directories are returned that are not Set Group ID up on execution (SGID) files and owned by a privileged account, this is a finding.

Fix

Configure Ubuntu 22.04 LTS commands directories to be protected from unauthorized access. Run the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin ! -group root -type d -exec chgrp root '{}' \;
```

V-260495

Title

Ubuntu 22.04 LTS must have system commands owned by "root" or a system account.

Description

If Ubuntu 22.04 LTS were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to Ubuntu 22.04 LTS with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the system commands contained in the following directories are owned by "root", or a required system account, by using the following command:

```
$ sudo find /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin ! -user root -type f -exec stat -c "%n %U" '{}' \;
```

If any system commands are returned and are not owned by a required system account, this is a finding.

Fix

Configure Ubuntu 22.04 LTS commands and their respective parent directories to be protected from unauthorized access. Run the following command, replacing "" with any system command not owned by "root" or a required system account:

```
$ sudo chown root <command_name>
```

V-260496

Title

Ubuntu 22.04 LTS must have system commands group-owned by "root" or a system account.

Description

If Ubuntu 22.04 LTS were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to Ubuntu 22.04 LTS with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the system commands contained in the following directories are group-owned by "root" or a required system account by using the following command:

```
$ sudo find -L /bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin ! -group root -type f ! -perm /2000 -exec stat -c "%n %G" '{}' \;
```

If any system commands are returned that are not Set Group ID upon execution (SGID) files and group-owned by a required system account, this is a finding.

Fix

Configure Ubuntu 22.04 LTS commands to be protected from unauthorized access.

Run the following command, replacing "" with any system command not group-owned by "root" or a required system account:

```
$ sudo chgrp root <command_name>
```

V-260497

Title

Ubuntu 22.04 LTS library directories must be owned by "root".

Description

If the operating system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to operating systems with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the systemwide shared library directories "/lib", "/lib64", and "/usr/lib" are owned by "root" by using the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -user root -type d -exec stat -c "%n %U" '{}' \;
```

If any systemwide library directory is returned, this is a finding.

Fix

Configure the library files and their respective parent directories to be protected from unauthorized access. Run the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -user root -type d -exec chown root '{}' \;
```

V-260498

Title

Ubuntu 22.04 LTS library directories must be group-owned by "root".

Description

If the operating system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to operating systems with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the systemwide library directories "/lib", "/lib64", and "/usr/lib" are group-owned by "root" by using the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -group root -type d -exec stat -c "%n %G" '{}' \;
```

If any systemwide shared library directory is returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS library directories to be protected from unauthorized access. Run the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -group root -type d -exec chgrp root '{}' \;
```

V-260499

Title

Ubuntu 22.04 LTS library files must be owned by "root".

Description

If the operating system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to operating systems with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the systemwide shared library files contained in the directories "/lib", "/lib64", and "/usr/lib" are owned by "root" by using the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -user root -type f -exec stat -c "%n %U" '{}' \;
```

If any systemwide library file is returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS library files to be protected from unauthorized access. Run the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -user root -type f -exec chown root '{}' \;
```

V-260500

Title

Ubuntu 22.04 LTS library files must be group-owned by "root".

Description

If the operating system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

This requirement applies to operating systems with software libraries that are accessible and configurable, as in the case of interpreted languages. Software libraries also include privileged programs that execute with escalated privileges. Only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Rule Check

Verify the systemwide library files contained in the directories "/lib", "/lib64", and "/usr/lib" are group-owned by "root", or a required system account, by using the following command:

```
$ sudo find /lib /usr/lib /lib64 ! -group root -type f -exec stat -c "%n %G" '{}' \;
```

If any systemwide shared library file is returned and is not group-owned by a required system account, this is a finding.

Fix

Configure Ubuntu 22.04 LTS library files to be protected from unauthorized access.

Run the following command, replacing "" with any system command not group-owned by "root" or a required system account:

```
$ sudo chgrp root <command_name>
```

V-260501

Title

Ubuntu 22.04 LTS must configure the directories used by the system journal to be owned by "root".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify the /run/log/journal and /var/log/journal directories are owned by "root" by using the following command:

```
$ sudo find /run/log/journal /var/log/journal  -type d -exec stat -c "%n %U" {} \;
/run/log/journal root
/var/log/journal root
/var/log/journal/3b018e681c904487b11671b9c1987cce root
```

If any output returned is not owned by "root", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to set the appropriate ownership to the directories used by the systemd journal:

Add or modify the following lines in the "/usr/lib/tmpfiles.d/systemd.conf" file:

z /run/log/journal 2640 root systemd-journal - - z /var/log/journal 2640 root systemd-journal - -

Restart the system for the changes to take effect.

V-260502

Title

Ubuntu 22.04 LTS must configure the directories used by the system journal to be group-owned by "systemd-journal".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify the /run/log/journal and /var/log/journal directories are group-owned by "systemd-journal" by using the following command:

```
$ sudo find /run/log/journal /var/log/journal  -type d -exec stat -c "%n %G" {} \;
/run/log/journal systemd-journal
/var/log/journal systemd-journal
/var/log/journal/3b018e681c904487b11671b9c1987cce systemd-journal
```

If any output returned is not group-owned by "systemd-journal", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to set the appropriate group-ownership to the directories used by the systemd journal:

Add or modify the following lines in the "/usr/lib/tmpfiles.d/systemd.conf" file:

z /run/log/journal 2640 root systemd-journal - - z /var/log/journal 2640 root systemd-journal - -

Restart the system for the changes to take effect.

V-260503

Title

Ubuntu 22.04 LTS must configure the files used by the system journal to be owned by "root".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify the /run/log/journal and /var/log/journal files are owned by "root" by using the following command:

```
$ sudo find /run/log/journal /var/log/journal  -type f -exec stat -c "%n %U" {} \;
/var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000003c7a-0006073f8d1c0fec.journal root
/var/log/journal/3b018e681c904487b11671b9c1987cce/system.journal root
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000.journal root
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000@bdedf14602ff4081a77dc7a6debc8626-00000000000062a6-00060b4b414b617a.journal root
/var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000005301-000609a409
```

593.journal root /var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000000001-000604dae53225ee.journal root
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000@bdedf14602ff4081a77dc7a6debc8626-000000000000083b-000604dae72c7e3b.journal root

If any output returned is not owned by "root", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to set the appropriate ownership to the files used by the systemd journal:

Add or modify the following lines in the "/usr/lib/tmpfiles.d/systemd.conf" file:

Z /run/log/journal/%m ~2640 root systemd-journal - - z /var/log/journal/%m 2640 root systemd-journal - - z /var/log/journal/%m/system.journal 0640 root systemd-journal - -

Restart the system for the changes to take effect.

V-260504

Title

Ubuntu 22.04 LTS must configure the files used by the system journal to be group-owned by "systemd-journal".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify the /run/log/journal and /var/log/journal files are group-owned by "systemd-journal" by using the following command:

```
$ sudo find /run/log/journal /var/log/journal  -type f -exec stat -c "%n %G" {} \;
/var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000003c7a-0006073f8d1c0fec.journal systemd-journal
/var/log/journal/3b018e681c904487b11671b9c1987cce/system.journal systemd-journal
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000.journal systemd-journal
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000@bdedf14602ff4081a77dc7a6debc8626-00000000000062a6-00060b4b414b617a.journal systemd-journal
/var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000005301-000609a409
```

593.journal systemd-journal /var/log/journal/3b018e681c904487b11671b9c1987cce/system@99dcc72bb1134aaeae4bf157aa7606f4-0000000000000001-000604dae53225ee.journal systemd-journal
/var/log/journal/3b018e681c904487b11671b9c1987cce/user-1000@bdedf14602ff4081a77dc7a6debc8626-000000000000083b-000604dae72c7e3b.journal systemd-journal

If any output returned is not group-owned by "systemd-journal", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to set the appropriate group-ownership to the files used by the systemd journal:

Add or modify the following line in the "/usr/lib/tmpfiles.d/systemd.conf" file:

Z /run/log/journal/%m ~2640 root systemd-journal - - z /var/log/journal/%m 2640 root systemd-journal - - z /var/log/journal/%m/system.journal 0640 root systemd-journal - -

Restart the system for the changes to take effect.

V-260505

Title

Ubuntu 22.04 LTS must be configured so that the "journalctl" command is owned by "root".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, Personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify that the "journalctl" command is owned by "root" by using the following command:

```
$ sudo find /usr/bin/journalctl -exec stat -c "%n %U" {} \;
/usr/bin/journalctl root
```

If "journalctl" is not owned by "root", this is a finding.

Fix

Configure "journalctl" to be owned by "root":

```
$ sudo chown root /usr/bin/journalctl
```

V-260506

Title

Ubuntu 22.04 LTS must be configured so that the "journalctl" command is group-owned by "root".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify that the "journalctl" command is group-owned by "root" by using the following command:

```
$ sudo find /usr/bin/journalctl -exec stat -c "%n %G" {} \;
/usr/bin/journalctl root
```

If "journalctl" is not group-owned by "root", this is a finding.

Fix

Configure "journalctl" to be group-owned by "root":

```
$ sudo chown :root /usr/bin/journalctl
```

V-260507

Title

Ubuntu 22.04 LTS must configure audit tools to be owned by "root".

Description

Protecting audit information also includes identifying and protecting the tools used to view and manipulate log data. Therefore, protecting audit tools is necessary to prevent unauthorized operation on audit information.

Operating systems providing tools to interface with audit information will leverage user permissions and roles identifying the user accessing the tools and the corresponding rights the user enjoys in order to make access decisions regarding the access to audit tools.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

Satisfies: SRG-OS-000256-GPOS-00097, SRG-OS-000257-GPOS-00098

Rule Check

Verify Ubuntu 22.04 LTS configures the audit tools to be owned by "root" to prevent any unauthorized access with the following command:

```
$ stat -c "%n %U" /sbin/auditctl /sbin/aureport /sbin/ausearch /sbin/autrace /sbin/auditd /sbin/audispd* /sbin/augenrules
/sbin/auditctl root
/sbin/aureport root
/sbin/ausearch root
/sbin/autrace root
/sbin/auditd root
/sbin/audispd-zos-remote root
/sbin/augenrules root
```

If any of the audit tools are not owned by "root", this is a finding.

Fix

Configure the audit tools on Ubuntu 22.04 LTS to be protected from unauthorized access by setting the file owner as root using the following command:

```
$ sudo chown root <audit_tool_name>
```

Replace "" with each audit tool not owned by "root".

V-260508

Title

Ubuntu 22.04 LTS must configure the "/var/log" directory to be owned by "root".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify Ubuntu 22.04 LTS configures the "/var/log" directory to be owned by "root" by using the following command:

```
$ stat -c "%n %U" /var/log
/var/log root
```

If the "/var/log" directory is not owned by "root", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to have root own the "/var/log" directory by using the following command:

```
$ sudo chown root /var/log
```

V-260509

Title

Ubuntu 22.04 LTS must configure the "/var/log" directory to be group-owned by "syslog".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify that Ubuntu 22.04 LTS configures the "/var/log" directory to be group-owned by "syslog" by using the following command:

```
$ stat -c "%n %G" /var/log
/var/log syslog
```

If the "/var/log" directory is not group-owned by "syslog", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to have syslog group-own the "/var/log" directory by using the following command:

```
$ sudo chgrp syslog /var/log
```

V-260510

Title

Ubuntu 22.04 LTS must configure "/var/log/syslog" file to be owned by "syslog".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify that Ubuntu 22.04 LTS configures the "/var/log/syslog" file to be owned by "syslog" by using the following command:

```
$ stat -c "%n %U" /var/log/syslog
/var/log/syslog
```

If the "/var/log/syslog" file is not owned by "syslog", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to have syslog own the "/var/log/syslog" file by using the following command:

```
$ sudo chown syslog /var/log/syslog
```

V-260511

Title

Ubuntu 22.04 LTS must configure the "/var/log/syslog" file to be group-owned by "adm".

Description

Only authorized personnel should be aware of errors and the details of the errors. Error messages are an indicator of an organization's operational state or can identify the operating system or platform. Additionally, personally identifiable information (PII) and operational information must not be revealed through error messages to unauthorized personnel or their designated representatives.

The structure and content of error messages must be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Rule Check

Verify that Ubuntu 22.04 LTS configures the "/var/log/syslog" file to be group-owned by "adm" by using the following command:

```
$ stat -c "%n %G" /var/log/syslog
/var/log/syslog adm
```

If the "/var/log/syslog" file is not group-owned by "adm", this is a finding.

Fix

Configure Ubuntu 22.04 LTS to have adm group-own the "/var/log/syslog" file by using the following command:

```
$ sudo chgrp adm /var/log/syslog
```

V-260512

Title

Ubuntu 22.04 LTS must be configured so that the "journalctl" command is not accessible by unauthorized users.

Description

Any operating system providing too much information in error messages risks compromising the data and security of the structure, and content of error messages needs to be carefully considered by the organization.

Organizations carefully consider the structure/content of error messages. The extent to which information systems are able to identify and handle error conditions is guided by organizational policy and operational requirements. Information that could be exploited by adversaries includes, for example, erroneous logon attempts with passwords entered by mistake as the username, mission/business information that can be derived from (if not stated explicitly by) information recorded, and personal information, such as account numbers, social security numbers, and credit card numbers.

Rule Check

Verify that the "journalctl" command has a permission set of "740" by using the following command:

```
$ sudo find /usr/bin/journalctl -exec stat -c "%n %a" {} \;
/usr/bin/journalctl 740
```

If "journalctl" is not set to "740", this is a finding.

Fix

Configure "journalctl" to have a permission set of "740":

```
$ sudo chmod 740 /usr/bin/journalctl
```

V-260513

Title

Ubuntu 22.04 LTS must set a sticky bit on all public directories to prevent unauthorized and unintended information transferred via shared system resources.

Description

Preventing unauthorized information transfers mitigates the risk of information, including encrypted representations of information, produced by the actions of prior users/roles (or the actions of processes acting on behalf of prior users/roles) from being available to any current users/roles (or current processes) that obtain access to shared system resources (e.g., registers, main memory, hard disks) after those resources have been released back to information systems. The control of information in shared resources is also commonly referred to as object reuse and residual information protection.

This requirement generally applies to the design of an information technology product, but it can also apply to the configuration of particular information system components that are, or use, such products. This can be verified by acceptance/validation processes in DOD or other government agencies.

There may be shared resources with configurable protections (e.g., files in storage) that may be assessed on specific information system components.

Rule Check

Verify that all public directories have the public sticky bit set by using the following command:

```
$ sudo find / -type d -perm -002 ! -perm -1000
```

If any public directories are found missing the sticky bit, this is a finding.

Fix

Configure all public directories to have the sticky bit set to prevent unauthorized and unintended information transferred via shared system resources.

Set the sticky bit on all public directories using the following command, replacing "" with any directory path missing the sticky bit:

```
$ sudo chmod +t  <public_directory_name>
```

V-260514

Title

Ubuntu 22.04 LTS must have an application firewall installed in order to control remote access methods.

Description

Remote access services, such as those providing remote access to network devices and information systems, which lack automated control capabilities, increase risk and make remote user access management difficult at best.

Remote access is access to DOD nonpublic information systems by an authorized user (or an information system) communicating through an external, nonorganization-controlled network. Remote access methods include, for example, dial-up, broadband, and wireless.

Ubuntu 22.04 LTS functionality (e.g., RDP) must be capable of taking enforcement action if the audit reveals unauthorized activity. Automated control of remote access sessions allows organizations to ensure ongoing compliance with remote access policies by enforcing connection rules of remote access applications on a variety of information system components (e.g., servers, workstations, notebook computers, smartphones, and tablets).

Rule Check

Verify that the Uncomplicated Firewall is installed by using the following command:

```
$ dpkg -l | grep ufw
ii     ufw     0.36.1-4ubuntu0.1     all     program for managing a Netfilter firewall
```

If the "ufw" package is not installed, ask the system administrator if another application firewall is installed.

If no application firewall is installed, this is a finding.

Fix

Install the Uncomplicated Firewall by using the following command:

```
$ sudo apt-get install ufw
```

V-260515

Title

Ubuntu 22.04 LTS must enable and run the Uncomplicated Firewall (ufw).

Description

Remote access services, such as those providing remote access to network devices and information systems, which lack automated control capabilities, increase risk and make remote user access management difficult at best.

Remote access is access to DOD nonpublic information systems by an authorized user (or an information system) communicating through an external, nonorganization-controlled network. Remote access methods include, for example, dial-up, broadband, and wireless.

Ubuntu 22.04 LTS functionality (e.g., RDP) must be capable of taking enforcement action if the audit reveals unauthorized activity. Automated control of remote access sessions allows organizations to ensure ongoing compliance with remote access policies by enforcing connection rules of remote access applications on a variety of information system components (e.g., servers, workstations, notebook computers, smartphones, and tablets).

Rule Check

Verify the ufw is enabled on the system with the following command:

```
$ sudo ufw status
Status: active
```

If the above command returns the status as "inactive" or any type of error, this is a finding.

Fix

Enable the ufw by using the following command:

```
$ sudo ufw enable
```

V-260516

Title

Ubuntu 22.04 LTS must have an application firewall enabled.

Description

Firewalls protect computers from network attacks by blocking or limiting access to open network ports. Application firewalls limit which applications are allowed to communicate over the network.

Rule Check

Verify the Uncomplicated Firewall (ufw) is enabled on the system with the following command:

```
$ systemctl status ufw.service | grep -i "active:"
Active: active (exited) since Thu 2022-12-25 00:00:01 NZTD; 365 days 11h ago
```

If "ufw.service" is "inactive", this is a finding.

If the ufw is not installed, ask the system administrator if another application firewall is installed. If no application firewall is installed, this is a finding.

Fix

Enable and start the ufw by using the following command:

```
$ sudo systemctl enable ufw.service --now
```

V-260517

Title

Ubuntu 22.04 LTS must configure the Uncomplicated Firewall (ufw) to rate-limit impacted network interfaces.

Description

Denial of service (DoS) is a condition when a resource is not available for legitimate users. When this occurs, the organization either cannot accomplish its mission or must operate at degraded capacity.

This requirement addresses the configuration of the operating system to mitigate the impact of DoS attacks that have occurred or are ongoing on system availability. For each system, known and potential DoS attacks must be identified and solutions for each type implemented. A variety of technologies exist to limit or, in some cases, eliminate the effects of DoS attacks (e.g., limiting processes or establishing memory partitions). Employing increased capacity and bandwidth, combined with service redundancy, may reduce the susceptibility to some DoS attacks.

Rule Check

Verify an application firewall is configured to rate limit any connection to the system.

Check all the services listening to the ports by using the following command:

```
$ ss -146ut
Netid               State          Recv-Q          Send-Q                                  Local Address:Port            Peer Address:Port               Process
tcp                 LISTEN               0                     511                                        *:http                       *:*
tcp                 LISTEN               0                     128                                     [::]:ssh                     [::]:*
tcp                 LISTEN               0                     128                                     [::]:ipp                     [::]:*
tcp                 LISTEN               0                     128                                     [::]:smtp                  [::]:*
```

For each entry, verify that the ufw is configured to rate limit the service ports by using the following command:

```
$ sudo ufw status
Status: active

To                         Action     From
--                         ------         ----
80/tcp                   LIMIT       Anywhere
25/tcp                   LIMIT       Anywhere
Anywhere            DENY       240.9.19.81
443                        LIMIT       Anywhere
22/tcp                   LIMIT      Anywhere
80/tcp (v6)            LIMIT      Anywhere
25/tcp (v6)            LIMIT      Anywhere
22/tcp (v6)            LIMIT      Anywhere (v6)

25                         DENY OUT    Anywhere
25 (v6)                  DENY OUT    Anywhere (v6)
```

If any port with a state of "LISTEN" that does not have an action of "DENY", is not marked with the "LIMIT" action, this is a finding.

Fix

Configure the application firewall to protect against or limit the effects of DoS attacks by ensuring Ubuntu 22.04 LTS is implementing rate-limiting measures on impacted network interfaces.

For each service with a port listening to connections, run the following command, replacing '" with the service that needs to be rate limited.

```
$ sudo ufw limit <service_name>
```

Rate-limiting can also be done on an interface. An example of adding a rate limit on the "ens160" interface follows:

```
$ sudo ufw limit in on ens160
```

V-260518

Title

Ubuntu 22.04 LTS must be configured to prohibit or restrict the use of functions, ports, protocols, and/or services, as defined in the PPSM CAL and vulnerability assessments.

Description

To prevent unauthorized connection of devices, unauthorized transfer of information, or unauthorized tunneling (i.e., embedding of data types within data types), organizations must disable or restrict unused or unnecessary physical and logical ports/protocols on information systems.

Operating systems are capable of providing a wide variety of functions and services. Some of the functions and services provided by default may not be necessary to support essential organizational operations. Additionally, it is sometimes convenient to provide multiple services from a single component (e.g., VPN and IPS); however, doing so increases risk over limiting the services provided by any one component.

To support the requirements and principles of least functionality, the operating system must support the organizational requirements, providing only essential capabilities and limiting the use of ports, protocols, and/or services to only those required, authorized, and approved to conduct official business or to address authorized quality of life issues.

Rule Check

Check the firewall configuration for any unnecessary or prohibited functions, ports, protocols, and/or services by using the following command:

```
$ sudo ufw show raw
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
     pkts      bytes target     prot opt in     out     source               destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
    pkts      bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
    pkts      bytes target     prot opt in     out     source               destination
```

Ask the system administrator for the site or program PPSM CLSA. Verify the services allowed by the firewall match the PPSM CLSA.

If there are any additional ports, protocols, or services that are not included in the PPSM CLSA, this is a finding.

If there are any ports, protocols, or services that are prohibited by the PPSM CAL, this is a finding.

Fix

Add all ports, protocols, or services allowed by the PPSM CLSA by using the following command:

```
$ sudo ufw allow <direction> <port/protocol/service>
```

Where the direction is "in" or "out" and the port is the one corresponding to the protocol or service allowed.

To deny access to ports, protocols, or services, use:

```
$ sudo ufw deny <direction> <port/protocol/service>
```

V-260519

Title

Ubuntu 22.04 LTS must, for networked systems, compare internal information system clocks at least every 24 hours with a server synchronized to one of the redundant United States Naval Observatory (USNO) time servers, or a time server designated for the appropriate DOD network (NIPRNet/SIPRNet), and/or the Global Positioning System (GPS).

Description

Inaccurate time stamps make it more difficult to correlate events and can lead to an inaccurate analysis. Determining the correct time a particular event occurred on a system is critical when conducting forensic analysis and investigating system events. Sources outside the configured acceptable allowance (drift) may be inaccurate.

Synchronizing internal information system clocks provides uniformity of time stamps for information systems with multiple system clocks and systems connected over a network.

Organizations should consider endpoints that may not have regular access to the authoritative time server (e.g., mobile, teleworking, and tactical endpoints).

Note that USNO offers authenticated NTP service to DOD and U.S. Government agencies operating on the NIPR and SIPR networks. Visit https://www.usno.navy.mil/USNO/time/ntp/DOD-customers for more information.

Rule Check

Verify Ubuntu 22.04 LTS is configured to compare the system clock at least every 24 hours to the authoritative time source by using the following command:

Note: If the system is not networked, this requirement is not applicable.

```
$ sudo grep maxpoll -ir /etc/chrony*
server tick.usno.navy.mil iburst maxpoll 16
```

If the "maxpoll" option is set to a number greater than 16, the line is commented out, or is missing, this is a finding.

Verify that the "chrony.conf" file is configured to an authoritative DOD time source by using the following command:

```
$ sudo grep -ir server /etc/chrony*
server tick.usno.navy.mil iburst maxpoll 16
server tock.usno.navy.mil iburst maxpoll 16
server ntp2.usno.navy.mil iburst maxpoll 16
```

If "server" is not defined, is not set to an authoritative DOD time source, is commented out, or missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to compare the system clock at least every 24 hours to the authoritative time source.

Add or modify the following line in the "/etc/chrony/chrony.conf" file:

server [source] iburst maxpoll = 16

Restart "chrony.service" for the changes to take effect by using the following command:

```
$ sudo systemctl restart chrony.service
```

V-260520

Title

Ubuntu 22.04 LTS must synchronize internal information system clocks to the authoritative time source when the time difference is greater than one second.

Description

Inaccurate time stamps make it more difficult to correlate events and can lead to an inaccurate analysis. Determining the correct time a particular event occurred on a system is critical when conducting forensic analysis and investigating system events.

Synchronizing internal information system clocks provides uniformity of time stamps for information systems with multiple system clocks and systems connected over a network. Organizations should consider setting time periods for different types of systems (e.g., financial, legal, or mission-critical systems).

Organizations should also consider endpoints that may not have regular access to the authoritative time server (e.g., mobile, teleworking, and tactical endpoints). This requirement is related to the comparison done every 24 hours in SRG-OS-000355 because a comparison must be done to determine the time difference.

Rule Check

Verify Ubuntu 22.04 LTS synchronizes internal system clocks to the authoritative time source when the time difference is greater than one second.

Note: If the system is not networked, this requirement is not applicable.

Check the value of "makestep" by using the following command:

```
$ grep -ir makestep /etc/chrony*
makestep 1 1
```

If "makestep" is not set to "1 1", is commented out, or is missing, this is a finding.

Verify the NTP service is active and the system clock is synchronized with the authoritative time source:

```
$ timedatectl | grep -Ei '(synchronized|service)'
System clock synchronized: yes
NTP service: active
```

If the NTP service is not active, this is a finding.

If the system clock is not synchronized, this is a finding.

Fix

Configure chrony to synchronize the internal system clocks to the authoritative source when the time difference is greater than one second by doing the following:

Edit the "/etc/chrony/chrony.conf" file and add:

```
makestep 1 1
```

Restart the chrony service:

```
$ sudo systemctl restart chrony.service
```

V-260521

Title

Ubuntu 22.04 LTS must record time stamps for audit records that can be mapped to Coordinated Universal Time (UTC).

Description

If time stamps are not consistently applied and there is no common time reference, it is difficult to perform forensic analysis.

Time stamps generated by the operating system include date and time. Time is commonly expressed in UTC or local time with an offset from UTC.

Rule Check

Verify the time zone is configured to use UTC by using the following command:

```
$ timedatectl status | grep -i "time zone"
Time zone: Etc/UTC (UTC, +0000)
```

If "Time zone" is not set to UTC, this is a finding.

Fix

To Configure Ubuntu 22.04 LTS time zone to use UTC, run the following command:

```
$ sudo timedatectl set-timezone Etc/UTC
```

V-260522

Title

Ubuntu 22.04 LTS must be configured to use TCP syncookies.

Description

DoS is a condition when a resource is not available for legitimate users. When this occurs, the organization either cannot accomplish its mission or must operate at degraded capacity.

Managing excess capacity ensures that sufficient capacity is available to counter flooding attacks. Employing increased capacity and service redundancy may reduce the susceptibility to some DoS attacks. Managing excess capacity may include, for example, establishing selected usage priorities, quotas, or partitioning.

Rule Check

Verify Ubuntu 22.04 LTS is configured to use TCP syncookies by using the following command:

```
$ sysctl net.ipv4.tcp_syncookies
net.ipv4.tcp_syncookies = 1
```

If the value is not "1", this is a finding.

Check the saved value of TCP syncookies by using the following command:

```
$ sudo grep -ir net.ipv4.tcp_syncookies /etc/sysctl.d/*.conf /run/sysctl.d/*.conf /usr/local/lib/sysctl.d/*.conf /usr/lib/sysctl.d/*.conf /lib/sysctl.d/*.conf /etc/sysctl.conf 2> /de
```

If the "net.ipv4.tcp_syncookies" option is not set to "1", is commented out, or is missing, this is a finding.

If conflicting results are returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to use TCP syncookies by using the following command:

```
$ sudo sysctl -w net.ipv4.tcp_syncookies = 1
```

If "1" is not the system's default value, add or update the following line in "/etc/sysctl.conf":

```
net.ipv4.tcp_syncookies = 1
```

V-260523

Title

Ubuntu 22.04 LTS must have SSH installed.

Description

Without protection of the transmitted information, confidentiality and integrity may be compromised because unprotected communications can be intercepted and either read or altered.

This requirement applies to both internal and external networks and all types of information system components from which information can be transmitted (e.g., servers, mobile devices, notebook computers, printers, copiers, scanners, and facsimile machines). Communication paths outside the physical protection of a controlled boundary are exposed to the possibility of interception and modification.

Protecting the confidentiality and integrity of organizational information can be accomplished by physical means (e.g., employing physical distribution systems) or by logical means (e.g., employing cryptographic techniques). If physical means of protection are employed, then logical means (cryptography) do not have to be employed, and vice versa.

Satisfies: SRG-OS-000423-GPOS-00187, SRG-OS-000425-GPOS-00189, SRG-OS-000426-GPOS-00190

Rule Check

Verify the SSH package is installed by using the following command:

```
$ sudo dpkg -l | grep openssh
ii     openssh-client     1:8.9p1-3ubuntu0.4     amd64     secure shell (SSH) client, for secure access to remote machines
ii  openssh-server     1:8.9p1-3ubuntu0.4     amd64     secure shell (SSH) server, for secure access from remote machines
ii  openssh-sftp-server     1:8.9p1-3ubuntu0.4     amd64     secure shell (SSH) sftp server module, for SFTP access from remote machines
```

If the "openssh" server package is not installed, this is a finding.

Fix

Install the "ssh" meta-package by using the following command:

```
$ sudo apt install ssh
```

V-260524

Title

Ubuntu 22.04 LTS must use SSH to protect the confidentiality and integrity of transmitted information.

Description

Without protection of the transmitted information, confidentiality and integrity may be compromised because unprotected communications can be intercepted and either read or altered.

This requirement applies to both internal and external networks and all types of information system components from which information can be transmitted (e.g., servers, mobile devices, notebook computers, printers, copiers, scanners, and facsimile machines). Communication paths outside the physical protection of a controlled boundary are exposed to the possibility of interception and modification.

Protecting the confidentiality and integrity of organizational information can be accomplished by physical means (e.g., employing physical distribution systems) or by logical means (e.g., employing cryptographic techniques). If physical means of protection are employed, then logical means (cryptography) do not have to be employed, and vice versa.

Satisfies: SRG-OS-000423-GPOS-00187, SRG-OS-000425-GPOS-00189, SRG-OS-000426-GPOS-00190

Rule Check

Verify the "ssh.service" is enabled and active by using the following commands:

```
$ sudo systemctl is-enabled ssh
enabled

$ sudo systemctl is-active ssh
active
```

If "ssh.service" is not enabled and active, this is a finding.

Fix

Enable and start the "ssh.service" by using the following command:

```
$ sudo systemctl enable ssh.service --now
```

V-260525

Title

Ubuntu 22.04 LTS must display the Standard Mandatory DOD Notice and Consent Banner before granting any local or remote connection to the system.

Description

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

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

The banner must be formatted in accordance with applicable DOD policy. Use the following verbiage for operating systems that can accommodate banners of 1300 characters:

"You are accessing a U.S. Government (USG) Information System (IS) that is provided for USG-authorized use only.

By using this IS (which includes any device attached to this IS), you consent to the following conditions:

-The USG routinely intercepts and monitors communications on this IS for purposes including, but not limited to, penetration testing, COMSEC monitoring, network operations and defense, personnel misconduct (PM), law enforcement (LE), and counterintelligence (CI) investigations.

-At any time, the USG may inspect and seize data stored on this IS.

-Communications using, or data stored on, this IS are not private, are subject to routine monitoring, interception, and search, and may be disclosed or used for any USG-authorized purpose.

-This IS includes security measures (e.g., authentication and access controls) to protect USG interests--not for your personal benefit or privacy.

-Notwithstanding the above, using this IS does not constitute consent to PM, LE or CI investigative searching or monitoring of the content of privileged communications, or work product, related to personal representation or services by attorneys, psychotherapists, or clergy, and their assistants. Such communications and work product are private and confidential. See User Agreement for details."

Use the following verbiage for operating systems that have severe limitations on the number of characters that can be displayed in the banner:

"I've read (literal ampersand) consent to terms in IS user agreem't."

Satisfies: SRG-OS-000023-GPOS-00006, SRG-OS-000228-GPOS-00088

Rule Check

Verify Ubuntu 22.04 LTS displays the Standard Mandatory DOD Notice and Consent Banner before granting access to Ubuntu 22.04 LTS via an SSH logon by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'banner'
/etc/ssh/sshd_config:Banner /etc/issue.net
```

The command will return the banner option along with the name of the file that contains the SSH banner. If the line is commented out, missing, or conflicting results are returned, this is a finding.

Verify the specified banner file matches the Standard Mandatory DOD Notice and Consent Banner exactly:

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

If the banner text does not match the Standard Mandatory DOD Notice and Consent Banner exactly, this is a finding.

Fix

Set the parameter Banner in "/etc/ssh/sshd_config" to point to the "/etc/issue.net" file:

```
$ sudo sed -i '/^Banner/d' /etc/ssh/sshd_config
$ sudo sed -i '$aBanner /etc/issue.net' /etc/ssh/sshd_config
```

Replace the text in "/etc/issue.net" with the Standard Mandatory DOD Notice and Consent Banner:

You are accessing a U.S. Government (USG) Information System (IS) that is provided for USG-authorized use only. By using this IS (which includes any device attached to this IS), you consent to the following conditions:

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

Restart the SSH daemon for the changes to take effect and then signal the SSH server to reload the configuration file:

```
$ sudo systemctl -s SIGHUP kill sshd
```

V-260526

Title

Ubuntu 22.04 LTS must not allow unattended or automatic login via SSH.

Description

Failure to restrict system access to authenticated users negatively impacts Ubuntu 22.04 LTS security.

Rule Check

Verify that unattended or automatic login via SSH is disabled by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iEH '(permit(.*?)(passwords|environment))'
/etc/ssh/sshd_config:PermitEmptyPasswords no
/etc/ssh/sshd_config:PermitUserEnvironment no
```

If "PermitEmptyPasswords" and "PermitUserEnvironment" are not set to "no", are commented out, are missing, or conflicting results are returned, this is a finding.

Fix

Configure the SSH server to not allow unattended or automatic login to the system.

Add or modify the following lines in the "/etc/ssh/sshd_config" file:

PermitEmptyPasswords no PermitUserEnvironment no

Restart the SSH daemon for the changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260527

Title

Ubuntu 22.04 LTS must be configured so that all network connections associated with SSH traffic terminate after becoming unresponsive.

Description

Terminating an unresponsive SSH session within a short time period reduces the window of opportunity for unauthorized personnel to take control of a management session enabled on the console or console port that has been left unattended. In addition, quickly terminating an idle SSH session will also free up resources committed by the managed network element.

Terminating network connections associated with communications sessions includes, for example, deallocating associated TCP/IP address/port pairs at the operating system level and deallocating networking assignments at the application level if multiple application sessions are using a single operating system-level network connection. This does not mean the operating system terminates all sessions or network access; it only ends the unresponsive session and releases the resources associated with that session.

Rule Check

Verify the SSH server automatically terminates a user session after the SSH client has become unresponsive by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'clientalivecountmax'
/etc/ssh/sshd_config:ClientAliveCountMax 1
```

If "ClientAliveCountMax" is not to "1", if conflicting results are returned, is commented out, or is missing, this is a finding.

Fix

Configure the SSH server to terminate a user session automatically after the SSH client has become unresponsive.

Note: This setting must be applied in conjunction with UBTU-22-255040 to function correctly.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

ClientAliveCountMax 1

Restart the SSH daemon for the changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260528

Title

Ubuntu 22.04 LTS must be configured so that all network connections associated with SSH traffic are terminated after 10 minutes of becoming unresponsive.

Description

Terminating an unresponsive SSH session within a short time period reduces the window of opportunity for unauthorized personnel to take control of a management session enabled on the console or console port that has been left unattended. In addition, quickly terminating an idle SSH session will also free up resources committed by the managed network element.

Terminating network connections associated with communications sessions includes, for example, deallocating associated TCP/IP address/port pairs at the operating system level and deallocating networking assignments at the application level if multiple application sessions are using a single operating system-level network connection. This does not mean that the operating system terminates all sessions or network access; it only ends the unresponsive session and releases the resources associated with that session.

Rule Check

Verify the SSH server automatically terminates a user session after the SSH client has been unresponsive for 10 minutes by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'clientaliveinterval'
/etc/ssh/sshd_config:ClientAliveInterval 600
```

If "ClientAliveInterval" does not exist, is not set to a value of "600" or less, if conflicting results are returned, is commented out, or is missing, this is a finding.

Fix

Configure the SSH server to terminate a user session automatically after the SSH client has been unresponsive for 10 minutes.

Note: This setting must be applied in conjunction with UBTU-22-255040 to function correctly.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

ClientAliveInterval 600

Restart the SSH daemon for the changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260529

Title

Ubuntu 22.04 LTS must be configured so that remote X connections are disabled, unless to fulfill documented and validated mission requirements.

Description

The security risk of using X11 forwarding is that the client's X11 display server may be exposed to attack when the SSH client requests forwarding. A system administrator may have a stance in which they want to protect clients that may expose themselves to attack by unwittingly requesting X11 forwarding, which can warrant a "no" setting.

X11 forwarding should be enabled with caution. Users with the ability to bypass file permissions on the remote host (for the user's X11 authorization database) can access the local X11 display through the forwarded connection. An attacker may then be able to perform activities such as keystroke monitoring if the ForwardX11Trusted option is also enabled.

If X11 services are not required for the system's intended function, they should be disabled or restricted as appropriate to the system's needs.

Rule Check

Verify that X11 forwarding is disabled by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'x11forwarding'
/etc/ssh/sshd_config:X11Forwarding no
```

If "X11Forwarding" is set to "yes" and is not documented with the information system security officer (ISSO) as an operational requirement, is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure the SSH server to disable X11 forwarding.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

X11Forwarding no

Restart the SSH daemon for the changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260530

Title

Ubuntu 22.04 LTS SSH daemon must prevent remote hosts from connecting to the proxy display.

Description

When X11 forwarding is enabled, there may be additional exposure to the server and client displays if the sshd proxy display is configured to listen on the wildcard address. By default, sshd binds the forwarding server to the loopback address and sets the hostname part of the DISPLAY environment variable to localhost. This prevents remote hosts from connecting to the proxy display.

Rule Check

Verify the SSH server prevents remote hosts from connecting to the proxy display by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'x11uselocalhost'
/etc/ssh/sshd_config:X11UseLocalhost yes
```

If "X11UseLocalhost" is set to "no", is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure the SSH server to prevent remote hosts from connecting to the proxy display.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

X11UseLocalhost yes

Restart the SSH daemon for the changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260531

Title

Ubuntu 22.04 LTS must configure the SSH daemon to use FIPSÂ 140-3-approved ciphers to prevent the unauthorized disclosure of information and/or detect changes to information during transmission.

Description

Without cryptographic integrity protections, information can be altered by unauthorized users without detection.

Remote access (e.g., RDP) is access to DOD nonpublic information systems by an authorized user (or an information system) communicating through an external, nonorganization-controlled network. Remote access methods include, for example, dial-up, broadband, and wireless.

Nonlocal maintenance and diagnostic activities are those activities conducted by individuals communicating through a network, either an external network (e.g., the internet) or an internal network.

Local maintenance and diagnostic activities are those activities carried out by individuals physically present at the information system or information system component and not communicating across a network connection.

Encrypting information for transmission protects information from unauthorized disclosure and modification. Cryptographic mechanisms implemented to protect information integrity include, for example, cryptographic hash functions which have common application in digital signatures, checksums, and message authentication codes.

By specifying a cipher list with the order of ciphers being in a "strongest to weakest" orientation, the system will automatically attempt to use the strongest cipher for securing SSH connections.

Satisfies: SRG-OS-000033-GPOS-00014, SRG-OS-000394-GPOS-00174, SRG-OS-000424-GPOS-00188

Rule Check

Verify the SSH server is configured to only implement FIPS-approved ciphers with the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'ciphers'
/etc/ssh/sshd_config:Ciphers aes256-ctr,aes256-gcm@openssh.com,aes192-ctr,aes128-ctr,aes128-gcm@openssh.com
```

If "Ciphers" does not contain only the ciphers "aes256-ctr,aes256-gcm@openssh.com,aes192-ctr,aes128-ctr,aes128-gcm@openssh.com" in exact order, is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure the SSH server to only implement FIPS-approved ciphers.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

Ciphers aes256-ctr,aes256-gcm@openssh.com,aes192-ctr,aes128-ctr,aes128-gcm@openssh.com

Restart the SSH server for the changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260532

Title

Ubuntu 22.04 LTS must configure the SSH daemon to use Message Authentication Codes (MACs) employing FIPS 140-3-approved cryptographic hashes to prevent the unauthorized disclosure of information and/or detect changes to information during transmission.

Description

Without cryptographic integrity protections, information can be altered by unauthorized users without detection.

Remote access (e.g., RDP) is access to DOD nonpublic information systems by an authorized user (or an information system) communicating through an external, non-organization-controlled network. Remote access methods include, for example, dial-up, broadband, and wireless. Nonlocal maintenance and diagnostic activities are those activities conducted by individuals communicating through a network, either an external network (e.g., the internet) or an internal network.

Local maintenance and diagnostic activities are those activities carried out by individuals physically present at the information system or information system component and not communicating across a network connection.

Encrypting information for transmission protects information from unauthorized disclosure and modification. Cryptographic mechanisms implemented to protect information integrity include, for example, cryptographic hash functions, which have common application in digital signatures, checksums, and message authentication codes.

Satisfies: SRG-OS-000250-GPOS-00093, SRG-OS-000393-GPOS-00173, SRG-OS-000424-GPOS-00188

Rule Check

Verify the SSH server is configured to only use MACs that employ FIPS 140-3 approved ciphers by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'macs'
```

```
/etc/ssh/sshd_config:MACs hmac-sha2-512,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-256-etm@openssh.com
```

If "MACs" does not contain only the hashes "hmac-sha2-512,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-256-etm@openssh.com" in exact order, is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure the SSH server to only use MACs that employ FIPS 140-3 approved hashes.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

MACs hmac-sha2-512,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-256-etm@openssh.com

Restart the SSH server for the changes to take effect:

```
$ sudo systemctl reload sshd.service
```

V-260533

Title

Ubuntu 22.04 LTS SSH server must be configured to use only FIPS-validated key exchange algorithms.

Description

Without cryptographic integrity protections provided by FIPS-validated cryptographic algorithms, information can be viewed and altered by unauthorized users without detection.

The system will attempt to use the first algorithm presented by the client that matches the server list. Listing the values "strongest to weakest" is a method to ensure the use of the strongest algorithm available to secure the SSH connection.

Rule Check

Verify that the SSH server is configured to use only FIPS-validated key exchange algorithms by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'kexalgorithms'
/etc/ssh/sshd_config:KexAlgorithms ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256
```

If "KexAlgorithms" does not contain only the algorithms "ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256" in exact order, is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure the SSH server to use only FIPS-validated key exchange algorithms.

Add or modify the following line in the "/etc/ssh/sshd_config" file:

KexAlgorithms ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256

Restart the SSH server for changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260534

Title

Ubuntu 22.04 LTS must use strong authenticators in establishing nonlocal maintenance and diagnostic sessions.

Description

Nonlocal maintenance and diagnostic activities are those activities conducted by individuals communicating through a network, either an external network (e.g., the internet) or an internal network. Local maintenance and diagnostic activities are those activities carried out by individuals physically present at the information system or information system component and not communicating across a network connection. Typically, strong authentication requires authenticators that are resistant to replay attacks and employ multifactor authentication. Strong authenticators include, for example, PKI where certificates are stored on a token protected by a password, passphrase, or biometric.

Rule Check

Verify Ubuntu 22.04 LTS is configured to use strong authenticators in the establishment of nonlocal maintenance and diagnostic maintenance by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'usepam'
/etc/ssh/sshd_config:UsePAM yes
```

If "UsePAM" is not set to "yes", is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to use strong authentication when establishing nonlocal maintenance and diagnostic sessions.

Add or modify the following line to /etc/ssh/sshd_config:

UsePAM yes

Restart the SSH server for changes to take effect:

```
$ sudo systemctl restart sshd.service
```

V-260535

Title

Ubuntu 22.04 LTS must enable the graphical user logon banner to display the Standard Mandatory DOD Notice and Consent Banner before granting local access to the system via a graphical user logon.

Description

Display of a standardized and approved use notification before granting access to Ubuntu 22.04 LTS ensures privacy and security notification verbiage used is consistent with applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance.

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

The banner must be formatted in accordance with applicable DOD policy. Use the following verbiage for operating systems that can accommodate banners of 1300 characters:

"You are accessing a U.S. Government (USG) Information System (IS) that is provided for USG-authorized use only.

By using this IS (which includes any device attached to this IS), you consent to the following conditions:

-The USG routinely intercepts and monitors communications on this IS for purposes including, but not limited to, penetration testing, COMSEC monitoring, network operations and defense, personnel misconduct (PM), law enforcement (LE), and counterintelligence (CI) investigations.

-At any time, the USG may inspect and seize data stored on this IS.

-Communications using, or data stored on, this IS are not private, are subject to routine monitoring, interception, and search, and may be disclosed or used for any USG-authorized purpose.

-This IS includes security measures (e.g., authentication and access controls) to protect USG interests--not for your personal benefit or privacy.

-Notwithstanding the above, using this IS does not constitute consent to PM, LE or CI investigative searching or monitoring of the content of privileged communications, or work product, related to personal representation or services by attorneys, psychotherapists, or clergy, and their assistants. Such communications and work product are private and confidential. See User Agreement for details."

Use the following verbiage for operating systems that have severe limitations on the number of characters that can be displayed in the banner:

"I've read (literal ampersand) consent to terms in IS user agreem't."

Rule Check

Verify Ubuntu 22.04 LTS is configured to display the Standard Mandatory DOD Notice and Consent Banner before granting access to the operating system via a graphical user logon by using the following command:

Note: If no graphical user interface is installed, this requirement is not applicable.

```
$ grep -i banner-message-enable /etc/gdm3/greeter.dconf-defaults
banner-message-enable=true
```

If the value for "banner-message-enable" is set to "false", the line is commented out, or no value is returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to display the Standard Mandatory DOD Notice and Consent Banner before granting access to the operating system via a graphical user logon.

Add or modify the following line in the "/etc/gdm3/greeter.dconf-defaults" file:

[org/gnome/login-screen]
banner-message-enable=true

Update GDM with the new configuration by using the following commands:

```
$ sudo dconf update
$ sudo systemctl restart gdm3
```

V-260536

Title

Ubuntu 22.04 LTS must display the Standard Mandatory DOD Notice and Consent Banner before granting local access to the system via a graphical user logon.

Description

Display of a standardized and approved use notification before granting access to Ubuntu 22.04 LTS ensures privacy and security notification verbiage used is consistent with applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance.

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

The banner must be formatted in accordance with applicable DOD policy. Use the following verbiage for operating systems that can accommodate banners of 1300 characters:

"You are accessing a U.S. Government (USG) Information System (IS) that is provided for USG-authorized use only.

By using this IS (which includes any device attached to this IS), you consent to the following conditions:

-The USG routinely intercepts and monitors communications on this IS for purposes including, but not limited to, penetration testing, COMSEC monitoring, network operations and defense, personnel misconduct (PM), law enforcement (LE), and counterintelligence (CI) investigations.

-At any time, the USG may inspect and seize data stored on this IS.

-Communications using, or data stored on, this IS are not private, are subject to routine monitoring, interception, and search, and may be disclosed or used for any USG-authorized purpose.

-This IS includes security measures (e.g., authentication and access controls) to protect USG interests--not for your personal benefit or privacy.

-Notwithstanding the above, using this IS does not constitute consent to PM, LE or CI investigative searching or monitoring of the content of privileged communications, or work product, related to personal representation or services by attorneys, psychotherapists, or clergy, and their assistants. Such communications and work product are private and confidential. See User Agreement for details."

Use the following verbiage for operating systems that have severe limitations on the number of characters that can be displayed in the banner:

"I've read (literal ampersand) consent to terms in IS user agreem't."

Rule Check

Verify Ubuntu 22.04 LTS displays the Standard Mandatory DOD Notice and Consent Banner before granting access to the operating system via a graphical user logon with the command:

Note: If no graphical user interface is installed, this requirement is not applicable.

```
$ grep -i banner-message-text /etc/gdm3/greeter.dconf-defaults
```

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

If the banner-message-text is missing, commented out, or does not match the Standard Mandatory DOD Notice and Consent Banner exactly, this is a finding.

Fix

Edit the "/etc/gdm3/greeter.dconf-defaults" file.

Set the "banner-message-text" line to contain the appropriate banner message text as shown below:

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

Update GDM with the new configuration by using the following commands:

```
$ sudo dconf update

$ sudo systemctl restart gdm3
```

V-260537

Title

Ubuntu 22.04 LTS must retain a user's session lock until that user reestablishes access using established identification and authentication procedures.

Description

A session lock is a temporary action taken when a user stops work and moves away from the immediate physical vicinity of the information system but does not want to log out because of the temporary nature of the absence.

The session lock is implemented at the point where session activity can be determined.

Regardless of where the session lock is determined and implemented, once invoked, a session lock of Ubuntu 22.04 LTS must remain in place until the user reauthenticates. No other activity aside from reauthentication must unlock the system.

Rule Check

Verify Ubuntu 22.04 LTS has a graphical user interface session lock enabled by using the following command:

Note: If no graphical user interface is installed, this requirement is not applicable.

```
$ sudo gsettings get org.gnome.desktop.screensaver lock-enabled
true
```

If "lock-enabled" is not set to "true", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to allow a user to lock the current graphical user interface session.

Set the "lock-enabled" setting to allow graphical user interface session locks by using the following command:

```
$ gsettings set org.gnome.desktop.screensaver lock-enabled true
```

V-260538

Title

Ubuntu 22.04 LTS must initiate a graphical session lock after 15 minutes of inactivity.

Description

A session lock is a temporary action taken when a user stops work and moves away from the immediate physical vicinity of the information system but does not want to log out because of the temporary nature of the absence.

The session lock is implemented at the point where session activity can be determined.

Regardless of where the session lock is determined and implemented, once invoked, a session lock of Ubuntu 22.04 LTS must remain in place until the user reauthenticates. No other activity aside from reauthentication must unlock the system.

Rule Check

Verify Ubuntu 22.04 LTS has a graphical user interface session lock configured to activate after 15 minutes of inactivity by using the following commands:

Note: If no graphical user interface is installed, this requirement is not applicable.

Get the following settings to verify the graphical user interface session is configured to lock the graphical user session after 15 minutes of inactivity:

```
$ gsettings get org.gnome.desktop.screensaver lock-enabled
true

$ gsettings get org.gnome.desktop.screensaver lock-delay
uint32 0

$ gsettings get org.gnome.desktop.session idle-delay
uint32 900
```

If "lock-enabled" is not set to "true", is commented out, or is missing, this is a finding.

If "lock-delay" is set to a value greater than "0", or if "idle-delay" is set to a value greater than "900", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to lock the current graphical user interface session after 15 minutes of inactivity.

Set the following settings to allow graphical user interface session lock to initiate after 15 minutes of inactivity:

```
$ gsettings set org.gnome.desktop.screensaver lock-enabled true

$ gsettings set org.gnome.desktop.screensaver lock-delay 0

$ gsettings set org.gnome.desktop.session idle-delay 900
```

V-260539

Title

Ubuntu 22.04 LTS must disable the x86 Ctrl-Alt-Delete key sequence if a graphical user interface is installed.

Description

A locally logged-on user who presses Ctrl-Alt-Delete, when at the console, can reboot the system. If accidentally pressed, as could happen in the case of a mixed OS environment, this can create the risk of short-term loss of availability of systems due to unintentional reboot. In the graphical environment, risk of unintentional reboot from the Ctrl-Alt-Delete sequence is reduced because the user will be prompted before any action is taken.

Rule Check

Verify Ubuntu 22.04 LTS is not configured to reboot the system when Ctrl-Alt-Delete is pressed when using a graphical user interface by using the following command:

Note: If no graphical user interface is installed, this requirement is not applicable.

```
$ gsettings get org.gnome.settings-daemon.plugins.media-keys logout
@as []
```

If the "logout" key is bound to an action, is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to disable the Ctrl-Alt-Delete sequence when using a graphical user interface.

```
$ gsettings set org.gnome.settings-daemon.plugins.media-keys logout []
```

Update the dconf settings:

```
# dconf update
```

V-260540

Title

Ubuntu 22.04 LTS must disable automatic mounting of Universal Serial Bus (USB) mass storage driver.

Description

Without authenticating devices, unidentified or unknown devices may be introduced, thereby facilitating malicious activity.

Peripherals include, but are not limited to, such devices as flash drives, external storage, and printers.

Rule Check

Verify Ubuntu 22.04 LTS disables ability to load the USB storage kernel module by using the following command:

```
$ grep usb-storage /etc/modprobe.d/* | grep "/bin/false"
/etc/modprobe.d/stig.conf:install usb-storage /bin/false
```

If the command does not return any output, or the line is commented out, this is a finding.

Verify Ubuntu 22.04 LTS disables the ability to use USB mass storage device.

```
$ grep usb-storage /etc/modprobe.d/* | grep -i "blacklist"
/etc/modprobe.d/stig.conf:blacklist usb-storage
```

If the command does not return any output, or the line is commented out, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to disable using the USB storage kernel module.

Create and/or append a custom file under "/etc/modprobe.d/" to contain the following:

```
$ sudo su -c "echo install usb-storage /bin/false >> /etc/modprobe.d/stig.conf"
```

Configure Ubuntu 22.04 LTS to disable the ability to use USB mass storage devices.

```
$ sudo su -c "echo blacklist usb-storage >> /etc/modprobe.d/stig.conf"
```

V-260541

Title

Ubuntu 22.04 LTS must disable all wireless network adapters.

Description

Without protection of communications with wireless peripherals, confidentiality and integrity may be compromised because unprotected communications can be intercepted and either read, altered, or used to compromise the operating system.

This requirement applies to wireless peripheral technologies (e.g., wireless mice, keyboards, displays, etc.) used with an operating system. Wireless peripherals (e.g., Wi-Fi/Bluetooth/IR Keyboards, Mice, and Pointing Devices and Near Field Communications [NFC]) present a unique challenge by creating an open, unsecured port on a computer. Wireless peripherals must meet DOD requirements for wireless data transmission and be approved for use by the AO. Even though some wireless peripherals, such as mice and pointing devices, do not ordinarily carry information that need to be protected, modification of communications with these wireless peripherals may be used to compromise the operating system. Communication paths outside the physical protection of a controlled boundary are exposed to the possibility of interception and modification.

Protecting the confidentiality and integrity of communications with wireless peripherals can be accomplished by physical means (e.g., employing physical barriers to wireless radio frequencies) or by logical means (e.g., employing cryptographic techniques). If physical means of protection are employed, then logical means (cryptography) do not have to be employed, and vice versa. If the wireless peripheral is only passing telemetry data, encryption of the data may not be required.

Rule Check

Verify that there are no wireless interfaces configured on the system by using the following command:

Note: If the system does not have any physical wireless network radios, this requirement is not applicable.

```
$ cat /proc/net/wireless
```

If any wireless interface names are listed under "Interface" and have not been documented and approved by the information system security officer (ISSO), this is a finding.

Fix

Disable all wireless network interfaces by using the following command:

```
$ sudo ifdown <wireless_interface_name>
```

For each interface listed, find their respective module by using the following command:

```
$ basename $(readlink -f /sys/class/net/<wireless_interface_name>/device/driver)
```

where must be substituted by the actual interface name.

Create and/or append a custom file under "/etc/modprobe.d/" by using the following command:

```
$ sudo su -c "echo install <module_name> /bin/false >> /etc/modprobe.d/stig.conf"
```

where must be substituted by the actual module name.

For each module from the system, execute the following command to remove it:

```
$ sudo modprobe -r <module_name>
```

V-260542

Title

Ubuntu 22.04 LTS must prevent direct login into the root account.

Description

To ensure individual accountability and prevent unauthorized access, organizational users must be individually identified and authenticated.

A group authenticator is a generic account used by multiple individuals. Use of a group authenticator alone does not uniquely identify individual users. Examples of the group authenticator is the Unix OS "root" user account, the Windows "Administrator" account, the "sa" account, or a "helpdesk" account.

For example, the Unix and Windows operating systems offer a "switch user" capability allowing users to authenticate with their individual credentials and, when needed, "switch" to the administrator role. This method provides for unique individual authentication prior to using a group authenticator.

Users (and any processes acting on behalf of users) must be uniquely identified and authenticated for all accesses other than those accesses explicitly identified and documented by the organization, which outlines specific user actions that can be performed on the operating system without identification or authentication.

Requiring individuals to be authenticated with an individual authenticator prior to using a group authenticator allows for traceability of actions, as well as adding an additional level of protection of the actions that can be taken with group account knowledge.

Rule Check

Verify Ubuntu 22.04 LTS prevents direct logins to the root account by using the following command:

```
$ sudo passwd -S root
root L 08/09/2022 0 99999 7 -1
```

If the output does not contain "L" in the second field to indicate the account is locked, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to prevent direct logins to the root account by using the following command:

```
$ sudo passwd -l root
```

V-260543

Title

Ubuntu 22.04 LTS must uniquely identify interactive users.

Description

To ensure accountability and prevent unauthenticated access, organizational users must be identified and authenticated to prevent potential misuse and compromise of the system.

Organizational users include organizational employees or individuals the organization deems to have equivalent status of employees (e.g., contractors). Organizational users (and processes acting on behalf of users) must be uniquely identified and authenticated to all accesses, except for the following:

1. Accesses explicitly identified and documented by the organization. Organizations document specific user actions that can be performed on the information system without identification or authentication; and
2. Accesses that occur through authorized use of group authenticators without individual authentication. Organizations may require unique identification of individuals in group accounts (e.g., shared privilege accounts) or for detailed accountability of individual activity.

Satisfies: SRG-OS-000104-GPOS-00051, SRG-OS-000121-GPOS-00062

Rule Check

Verify Ubuntu 22.04 LTS contains no duplicate User IDs (UIDs) for interactive users by using the following command:

```
$ awk -F ":" 'list[$3]++{print $1, $3}' /etc/passwd
```

If output is produced and the accounts listed are interactive user accounts, this is a finding.

Fix

Edit the file "/etc/passwd" and provide each interactive user account that has a duplicate UID with a unique UID.

V-260545

Title

Ubuntu 22.04 LTS must enforce 24 hours/one day as the minimum password lifetime. Passwords for new users must have a 24 hours/one day minimum password lifetime restriction.

Description

Enforcing a minimum password lifetime helps to prevent repeated password changes to defeat the password reuse or history enforcement requirement. If users are allowed to immediately and continually change their password, then the password could be repeatedly changed in a short period of time to defeat the organization's policy regarding password reuse.

Rule Check

Verify Ubuntu 22.04 LTS enforces a 24 hours/one day minimum password lifetime for new user accounts by using the following command:

```
$ grep -i pass_min_days /etc/login.defs
PASS_MIN_DAYS    1
```

If "PASS_MIN_DAYS" is less than "1", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce a 24 hours/one day minimum password lifetime.

Add or modify the following line in the "/etc/login.defs" file:

PASS_MIN_DAYS 1

V-260546

Title

Ubuntu 22.04 LTS must enforce a 60-day maximum password lifetime restriction. Passwords for new users must have a 60-day maximum password lifetime restriction.

Description

Any password, no matter how complex, can eventually be cracked. Therefore, passwords need to be changed periodically. If the operating system does not limit the lifetime of passwords and force users to change their passwords, there is the risk that the operating system passwords could be compromised.

Rule Check

Verify Ubuntu 22.04 LTS enforces a 60-day maximum password lifetime for new user accounts by using the following command:

```
$ grep -i pass_max_days /etc/login.defs
PASS_MAX_DAYS    60
```

If "PASS_MAX_DAYS" is less than "60", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce a 60-day maximum password lifetime.

Add or modify the following line in the "/etc/login.defs" file:

PASS_MAX_DAYS 60

V-260547

Title

Ubuntu 22.04 LTS must disable account identifiers (individuals, groups, roles, and devices) after 35 days of inactivity.

Description

Inactive identifiers pose a risk to systems and applications because attackers may exploit an inactive identifier and potentially obtain undetected access to the system. Owners of inactive accounts will not notice if unauthorized access to their user account has been obtained.

Operating systems need to track periods of inactivity and disable application identifiers after 35 days of inactivity.

Rule Check

Verify the account identifiers (individuals, groups, roles, and devices) are disabled after 35 days of inactivity by using the following command:

Check the account inactivity value by performing the following command:

```
$ grep INACTIVE /etc/default/useradd
INACTIVE=35
```

If "INACTIVE" is set to "-1" or is not set to "35", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to disable account identifiers after 35 days of inactivity after the password expiration.

Run the following command to change the configuration for adduser:

```
$ sudo useradd -D -f 35
```

Note: DOD recommendation is 35 days, but a lower value is acceptable. The value "0" will disable the account immediately after the password expires.

V-260548

Title

Ubuntu 22.04 LTS must automatically expire temporary accounts within 72 hours.

Description

Temporary accounts are privileged or nonprivileged accounts established during pressing circumstances, such as new software or hardware configuration or an incident response, where the need for prompt account activation requires bypassing normal account authorization procedures. If any inactive temporary accounts are left enabled on the system and are not either manually removed or automatically expired within 72 hours, the security posture of the system will be degraded and exposed to exploitation by unauthorized users or insider threat actors.

Temporary accounts are different from emergency accounts. Emergency accounts, also known as "last resort" or "break glass" accounts, are local logon accounts enabled on the system for emergency use by authorized system administrators to manage a system when standard logon methods are failing or not available. Emergency accounts are not subject to manual removal or scheduled expiration requirements.

The automatic expiration of temporary accounts may be extended as needed by the circumstances, but it must not be extended indefinitely. A documented permanent account should be established for privileged users who need long-term maintenance accounts.

Satisfies: SRG-OS-000002-GPOS-00002, SRG-OS-000123-GPOS-00064

Rule Check

Verify temporary accounts have been provisioned with an expiration date of 72 hours by using the following command:

```
$ sudo chage -l <temporary_account_name> | grep -E '(Password|Account) expires'
Password expires     : Apr 1, 2024
Account expires      : Apr 1, 2024
```

Verify each of these accounts has an expiration date set within 72 hours.

If any temporary accounts have no expiration date set or do not expire within 72 hours, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to expire temporary accounts after 72 hours by using the following command:

```
$ sudo chage -E $(date -d +3days +%Y-%m-%d) <temporary_account_name>
```

V-260549

Title

Ubuntu 22.04 LTS must automatically lock an account until the locked account is released by an administrator when three unsuccessful logon attempts have been made.

Description

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

Satisfies: SRG-OS-000021-GPOS-00005, SRG-OS-000329-GPOS-00128

Rule Check

Verify that Ubuntu 22.04 LTS utilizes the "pam_faillock" module by using the following command:

```
$ grep faillock /etc/pam.d/common-auth
```

auth [default=die] pam_faillock.so authfail auth sufficient pam_faillock.so authsucc

If the "pam_faillock.so" module is not present in the "/etc/pam.d/common-auth" file, this is a finding.

Verify the "pam_faillock" module is configured to use the following options:

```
$ sudo grep -Ew 'silent|audit|deny|fail_interval|unlock_time' /etc/security/faillock.conf
audit
silent
deny = 3
fail_interval = 900
unlock_time = 0
```

If "audit" is commented out, or is missing, this is a finding.

If "silent" is commented out, or is missing, this is a finding.

If "deny" is set to a value greater than "3", is commented out, or is missing, this is a finding.

If "fail_interval" is set to a value greater than "900", is commented out, or is missing, this is a finding.

If "unlock_time" is not set to "0", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to utilize the "pam_faillock" module.

Add or modify the following lines in the "/etc/pam.d/common-auth" file, below the "auth" definition for "pam_unix.so":

auth [default=die] pam_faillock.so authfail auth sufficient pam_faillock.so authsucc

Configure the "pam_faillock" module to use the following options.

Add or modify the following lines in the "/etc/security/faillock.conf" file:

audit silent deny = 3 fail_interval = 900 unlock_time = 0

V-260550

Title

Ubuntu 22.04 LTS must enforce a delay of at least four seconds between logon prompts following a failed logon attempt.

Description

Limiting the number of logon attempts over a certain time interval reduces the chances that an unauthorized user may gain access to an account.

Rule Check

Verify Ubuntu 22.04 LTS enforces a delay of at least four seconds between logon prompts following a failed logon attempt by using the following command:

```
$ grep pam_faildelay /etc/pam.d/common-auth
auth     required     pam_faildelay.so     delay=4000000
```

If "delay" is not set to "4000000" or greater, the line is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce a delay of at least four seconds between logon prompts following a failed logon attempt.

Add or modify the following line in the "/etc/pam.d/common-auth" file:

auth required pam_faildelay.so delay=4000000

V-260551

Title

Ubuntu 22.04 LTS must display the date and time of the last successful account logon upon logon.

Description

Configuration settings are the set of parameters that can be changed in hardware, software, or firmware components of the system that affect the security posture and/or functionality of the system. Security-related parameters are those parameters impacting the security state of the system, including the parameters required to satisfy other security control requirements. Security-related parameters include, for example: registry settings; account, file, directory permission settings; and settings for functions, ports, protocols, services, and remote connections.

Rule Check

Verify users are provided with feedback on when account accesses last occurred by using the following command:

```
$ grep pam_lastlog /etc/pam.d/login
session     required     pam_lastlog.so     showfailed
```

If the line containing "pam_lastlog" is not set to "required", or the "silent" option is present, the "showfailed" option is missing, the line is commented out, or the line is missing , this is a finding.

Fix

Configure Ubuntu 22.04 LTS to provide users with feedback on when account accesses last occurred.

Add or modify the following line at the top in the "/etc/pam.d/login" file:

session required pam_lastlog.so showfailed

V-260552

Title

Ubuntu 22.04 LTS must limit the number of concurrent sessions to ten for all accounts and/or account types.

Description

Ubuntu 22.04 LTS management includes the ability to control the number of users and user sessions that utilize an operating system. Limiting the number of allowed users and sessions per user is helpful in reducing the risks related to denial-of-service (DoS) attacks.

This requirement addresses concurrent sessions for information system accounts and does not address concurrent sessions by single users via multiple system accounts. The maximum number of concurrent sessions should be defined based upon mission needs and the operational environment for each system.

Rule Check

Verify Ubuntu 22.04 LTS limits the number of concurrent sessions to 10 for all accounts and/or account types by using the following command:

```
$ sudo grep -r -s '^[^#].*maxlogins' /etc/security/limits.conf /etc/security/limits.d/*.conf
/etc/security/limits.conf:* hard maxlogins 10
```

If "maxlogins" does not have a value of "10" or less, is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to limit the number of concurrent sessions to 10 for all accounts and/or account types.

Add or modify the following line at the top of the "/etc/security/limits.conf" file:

- hard maxlogins 10

V-260553

Title

Ubuntu 22.04 LTS must allow users to directly initiate a session lock for all connection types.

Description

A session lock is a temporary action taken when a user stops work and moves away from the immediate physical vicinity of the information system but does not want to log out because of the temporary nature of the absence.

The session lock is implemented at the point where session activity can be determined. Rather than be forced to wait for a period of time to expire before the user session can be locked, Ubuntu 22.04 LTS need to provide users with the ability to manually invoke a session lock so users may secure their session if they need to temporarily vacate the immediate physical vicinity.

Satisfies: SRG-OS-000030-GPOS-00011, SRG-OS-000031-GPOS-00012

Rule Check

Verify Ubuntu 22.04 LTS has the "vlock" package installed by using the following command:

```
$ dpkg -l | grep vlock
ii     vlock     2.2.2-10     amd64     Virtual Console locking program
```

If "vlock" is not installed, this is a finding.

Fix

Install the "vlock" package by using the following command:

```
$ sudo apt-get install vlock
```

V-260554

Title

Ubuntu 22.04 LTS must automatically exit interactive command shell user sessions after 15 minutes of inactivity.

Description

Terminating an idle interactive command shell user session within a short time period reduces the window of opportunity for unauthorized personnel to take control of it when left unattended in a virtual terminal or physical console.

Rule Check

Verify Ubuntu 22.04 LTS is configured to automatically exit interactive command shell user sessions after 15 minutes of inactivity or less by using the following command:

```
$ sudo grep -E "\bTMOUT=[0-9]+" /etc/bash.bashrc /etc/profile.d/*
/etc/profile.d/99-terminal_tmout.sh:TMOUT=900
```

If "TMOUT" is not set to "900" or less, is set to "0", is commented out, or missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to exit interactive command shell user sessions after 15 minutes of inactivity.

Create and/or append a custom file under "/etc/profile.d/" by using the following command:

```
$ sudo su -c "echo TMOUT=900 >> /etc/profile.d/99-terminal_tmout.sh"
```

This will set a timeout value of 15 minutes for all future sessions.

To set the timeout for the current sessions, execute the following command over the terminal session:

```
$ export TMOUT=900
```

V-260555

Title

Ubuntu 22.04 LTS default filesystem permissions must be defined in such a way that all authenticated users can read and modify only their own files.

Description

Setting the most restrictive default permissions ensures newly created accounts do not have unnecessary access.

Rule Check

Verify Ubuntu 22.04 LTS defines default permissions for all authenticated users in such a way that the user can read and modify only their own files by using the following command:

```
$ grep -i '^\s*umask' /etc/login.defs
UMASK 077
```

If the "UMASK" variable is set to "000", this is a finding with the severity raised to a CAT I.

If "UMASK" is not set to "077", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to define the default permissions for all authenticated users in such a way that the user can read and modify only their own files.

Add or modify the following line in the "/etc/login.defs" file:

UMASK 077

V-260556

Title

Ubuntu 22.04 LTS must have the "apparmor" package installed.

Description

Control of program execution is a mechanism used to prevent execution of unauthorized programs. Some operating systems may provide a capability that runs counter to the mission or provides users with functionality that exceeds mission requirements. This includes functions and services installed at the operating system level.

Some of the programs, installed by default, may be harmful or may not be necessary to support essential organizational operations (e.g., key missions, functions). Removal of executable programs is not always possible; therefore, establishing a method of preventing program execution is critical to maintaining a secure system baseline.

Methods for complying with this requirement include restricting execution of programs in certain environments, while preventing execution in other environments; or limiting execution of certain program functionality based on organization-defined criteria (e.g., privileges, subnets, sandboxed environments, or roles).

Satisfies: SRG-OS-000312-GPOS-00124, SRG-OS-000368-GPOS-00154, SRG-OS-000370-GPOS-00155

Rule Check

Verify Ubuntu 22.04 LTS has the "apparmor" package installed by using the following command:

```
$ dpkg -l | grep apparmor
ii     apparmor     3.0.4-2ubuntu2.3     amd64     user-space parser utility for AppArmor
```

If the "apparmor" package is not installed,Â this is a finding.

Fix

Install the "appArmor" package by using the following command:

```
$ sudo apt-get install apparmor
```

V-260557

Title

Ubuntu 22.04 LTS must be configured to use AppArmor.

Description

Control of program execution is a mechanism used to prevent execution of unauthorized programs. Some operating systems may provide a capability that runs counter to the mission or provides users with functionality that exceeds mission requirements. This includes functions and services installed at the operating system level.

Some of the programs, installed by default, may be harmful or may not be necessary to support essential organizational operations (e.g., key missions, functions). Removal of executable programs is not always possible; therefore, establishing a method of preventing program execution is critical to maintaining a secure system baseline.

Methods for complying with this requirement include restricting execution of programs in certain environments, while preventing execution in other environments; or limiting execution of certain program functionality based on organization-defined criteria (e.g., privileges, subnets, sandboxed environments, or roles).

Satisfies: SRG-OS-000368-GPOS-00154, SRG-OS-000370-GPOS-00155, SRG-OS-000324-GPOS-00125

Rule Check

Verify Ubuntu 22.04 LTS AppArmor is active by using the following commands:

```
$ systemctl is-enabled apparmor.service
enabled

$ systemctl is-active apparmor.service
active
```

If "apparmor.service" is not enabled and active, this is a finding.

Check if AppArmor profiles are loaded and enforced by using the following command:

```
$ sudo apparmor_status | grep -i profile
32 profiles are loaded.
32 profiles are in enforce mode.
0 profiles are in complain mode.
0 profiles are in kill mode.
0 profiles are in unconfined mode.
2 processes have profiles defined.
0 processes are unconfined but have a profile defined.
```

If no profiles are loaded and enforced, this is a finding.

Fix

Enable and start "apparmor.service" by using the following command:

```
$ sudo systemctl enable apparmor.service --now
```

Note: AppArmor must have properly configured profiles for applications and home directories. All configurations will be based on the actual system setup and organization and normally are on a per role basis. See the AppArmor documentation for more information on configuring profiles.

V-260558

Title

Ubuntu 22.04 LTS must require users to reauthenticate for privilege escalation or when changing roles.

Description

Without reauthentication, 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 reauthenticate.

Satisfies: SRG-OS-000373-GPOS-00156, SRG-OS-000373-GPOS-00157

Rule Check

Verify the "/etc/sudoers" file has no occurrences of "NOPASSWD" or "!authenticate" by using the following command:

```
$ sudo grep -Ei '(nopasswd|!authenticate)' /etc/sudoers /etc/sudoers.d/*
```

If any occurrences of "NOPASSWD" or "!authenticate" return from the command, this is a finding.

Fix

Remove any occurrence of "NOPASSWD" or "!authenticate" found in "/etc/sudoers" file or files in the "/etc/sudoers.d" directory.

V-260559

Title

Ubuntu 22.04 LTS must ensure only users who need access to security functions are part of sudo group.

Description

An isolation boundary provides access control and protects the integrity of the hardware, software, and firmware that perform security functions.

Security functions are the hardware, software, and/or firmware of the information system responsible for enforcing the system security policy and supporting the isolation of code and data on which the protection is based. Operating systems implement code separation (i.e., separation of security functions from nonsecurity functions) in a number of ways, including through the provision of security kernels via processor rings or processor modes. For nonkernel code, security function isolation is often achieved through file system protections that serve to protect the code on disk and address space protections that protect executing code.

Developers and implementers can increase the assurance in security functions by employing well-defined security policy models; structured, disciplined, and rigorous hardware and software development techniques; and sound system/security engineering principles. Implementation may include isolation of memory space and libraries.

Ubuntu 22.04 LTS restricts access to security functions through the use of access control mechanisms and by implementing least privilege capabilities.

Rule Check

Verify the sudo group has only members who require access to security functions by using the following command:

```
$ grep sudo /etc/group
sudo:x:27:<username>
```

If the sudo group contains users not needing access to security functions, this is a finding.

Fix

Configure the sudo group with only members requiring access to security functions.

To remove a user from the sudo group, run:

```
$ sudo gpasswd -d <username> sudo
```

V-260560

Title

Ubuntu 22.04 LTS must enforce password complexity by requiring at least one uppercase character be used.

Description

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

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

Rule Check

Verify Ubuntu 22.04 LTS enforces password complexity by requiring at least one uppercase character be used by using the following command:

```
$ grep -i ucredit /etc/security/pwquality.conf
ucredit = -1
```

If "ucredit" is greater than "-1", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce password complexity by requiring that at least one uppercase character be used.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

ucredit = -1

V-260561

Title

Ubuntu 22.04 LTS must enforce password complexity by requiring at least one lowercase character be used.

Description

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

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

Rule Check

Verify Ubuntu 22.04 LTS enforces password complexity by requiring that at least one lowercase character be used by using the following command:

```
$ grep -i lcredit /etc/security/pwquality.conf
lcredit = -1
```

If "lcredit" is greater than "-1", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce password complexity by requiring that at least one lowercase character be used.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

lcredit = -1

V-260562

Title

Ubuntu 22.04 LTS must enforce password complexity by requiring that at least one numeric character be used.

Description

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

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

Rule Check

Verify Ubuntu 22.04 LTS enforces password complexity by requiring that at least one numeric character be used by using the following command:

```
$ grep -i dcredit /etc/security/pwquality.conf
dcredit = -1
```

If "dcredit" is greater than "-1", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce password complexity by requiring that at least one numeric character be used.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

dcredit = -1

V-260563

Title

Ubuntu 22.04 LTS must enforce password complexity by requiring that at least one special character be used.

Description

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

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

Special characters are those characters that are not alphanumeric. Examples include: ~ ! @ # $ % ^ *.

Rule Check

Verify Ubuntu 22.04 LTS enforces password complexity by requiring that at least one special character be used by using the following command:

```
$ grep -i ocredit /etc/security/pwquality.conf
ocredit = -1
```

If "ocredit" is greater than "-1", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce password complexity by requiring that at least one special character be used.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

ocredit = -1

V-260564

Title

Ubuntu 22.04 LTS must prevent the use of dictionary words for passwords.

Description

If Ubuntu 22.04 LTS allows the user to select passwords based on dictionary words, then this increases the chances of password compromise by increasing the opportunity for successful guesses and brute-force attacks.

Rule Check

Verify Ubuntu 22.04 LTS prevents the use of dictionary words for passwords by using the following command:

```
$ grep -i dictcheck /etc/security/pwquality.conf
dictcheck = 1
```

If "dictcheck" is not set to "1", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to prevent the use of dictionary words for passwords.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

dictcheck = 1

V-260565

Title

Ubuntu 22.04 LTS must enforce a minimum 15-character password length.

Description

The shorter the password, the lower the number of possible combinations that need to be tested before the password is compromised.

Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks. Password length is one factor of several that helps to determine strength and how long it takes to crack a password. Use of more characters in a password helps to exponentially increase the time and/or resources required to compromise the password.

Rule Check

Verify the pwquality configuration file enforces a minimum 15-character password length by using the following command:

```
$ grep -i minlen /etc/security/pwquality.conf
minlen = 15
```

If "minlen" is not "15" or higher, is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce a minimum 15-character password length.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

minlen = 15

V-260566

Title

Ubuntu 22.04 LTS must require the change of at least eight characters when passwords are changed.

Description

If the operating system allows the user to consecutively reuse extensive portions of passwords, this increases the chances of password compromise by increasing the window of opportunity for attempts at guessing and brute-force attacks.

The number of changed characters refers to the number of changes required with respect to the total number of positions in the current password. In other words, characters may be the same within the two passwords; however, the positions of the like characters must be different.

If the password length is an odd number then number of changed characters must be rounded up. For example, a password length of 15 characters must require the change of at least eight characters.

Rule Check

Verify Ubuntu 22.04 LTS requires the change of at least eight characters when passwords are changed by using the following command:

```
$ grep -i difok /etc/security/pwquality.conf
difok = 8
```

If "difok" is less than "8", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to require the change of at least eight characters when passwords are changed.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

difok = 8

V-260567

Title

Ubuntu 22.04 LTS must be configured so that when passwords are changed or new passwords are established, pwquality must be used.

Description

Use of a complex password helps to increase the time and resources required to compromise the password. Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks. "pwquality" enforces complex password construction configuration and has the ability to limit brute-force attacks on the system.

Rule Check

Verify Ubuntu 22.04 LTS enforces password complexity rules by using the following command:

```
$ grep -i enforcing /etc/security/pwquality.conf
enforcing = 1
```

If "enforcing" is not "1", is commented out, or is missing, this is a finding.

Check for the use of "pwquality" by using the following command:

```
$ cat /etc/pam.d/common-password | grep requisite | grep pam_pwquality
 password     requisite     pam_pwquality.so retry=3
```

If "retry" is set to "0" or is greater than "3", or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to enforce password complexity rules.

Add or modify the following line in the "/etc/security/pwquality.conf" file:

enforcing = 1

Add or modify the following line in the "/etc/pam.d/common-password" file:

password requisite pam_pwquality.so retry=3

Note: The value of "retry" should be between "1" and "3".

V-260569

Title

Ubuntu 22.04 LTS must store only encrypted representations of passwords.

Description

Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks. If the information system or application allows the user to consecutively reuse their password when that password has exceeded its defined lifetime, the end result is a password that is not changed as per policy requirements.

Rule Check

Verify the Ubuntu operating stores only encrypted representations of passwords with the following command:

```
$ grep pam_unix.so /etc/pam.d/common-password
password [success=1 default=ignore] pam_unix.so obscure sha512 shadow remember=5 rounds=5000
```

If "sha512" is missing from the "pam_unix.so" line, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to store encrypted representations of passwords.

Add or modify the following line in the "/etc/pam.d/common-password" file:

password [success=1 default=ignore] pam_unix.so obscure sha512 shadow remember=5 rounds=5000

V-260570

Title

Ubuntu 22.04 LTS must not allow accounts configured with blank or null passwords.

Description

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

Rule Check

To verify that null passwords cannot be used, run the following command:

```
$ grep nullok /etc/pam.d/common-password
```

If this produces any output, this is a finding.

Fix

Remove any instances of the "nullok" option in "/etc/pam.d/common-password" to prevent logons with empty passwords.

V-260571

Title

Ubuntu 22.04 LTS must not have accounts configured with blank or null passwords.

Description

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

Rule Check

Verify all accounts on the system to have a password by using the following command:

```
$ sudo awk -F: '!$2 {print $1}' /etc/shadow
```

If the command returns any results, this is a finding.

Fix

Configure all accounts on the system to have a password or lock the account by using the following commands:

Set the account password:

```
$ sudo passwd <username>
```

Or lock the account:

```
$ sudo passwd -l <username>
```

V-260572

Title

Ubuntu 22.04 LTS must encrypt all stored passwords with a FIPS 140-3-approved cryptographic hashing algorithm.

Description

Passwords need to be protected at all times, and encryption is the standard method for protecting passwords. If passwords are not encrypted, they can be plainly read (i.e., clear text) and easily compromised.

Rule Check

Verify that the shadow password suite configuration is set to encrypt passwords with a FIPS 140-3 approved cryptographic hashing algorithm by using the following command:

```
$ grep -i '^\s*encrypt_method' /etc/login.defs
ENCRYPT_METHOD SHA512
```

If "ENCRYPT_METHOD" does not equal SHA512 or greater, is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to encrypt all stored passwords.

Add or modify the following line in the "/etc/login.defs" file:

ENCRYPT_METHOD SHA512

V-260573

Title

Ubuntu 22.04 LTS must implement multifactor authentication for remote access to privileged accounts in such a way that one of the factors is provided by a device separate from the system gaining access.

Description

Using an authentication device, such as a CAC or token separate from the information system, ensures that even if the information system is compromised, that compromise will not affect credentials stored on the authentication device.

Multifactor solutions that require devices separate from information systems gaining access include, for example, hardware tokens providing time-based or challenge-response authenticators and smart cards such as the U.S. Government personal identity verification card and the DOD common access card.

A privileged account is defined as an information system account with authorizations of a privileged user.

Remote access is access to DOD nonpublic information systems by an authorized user (or an information system) communicating through an external, nonorganization-controlled network. Remote access methods include, for example, dial-up, broadband, and wireless.

This requirement only applies to components where this is specific to the function of the device or has the concept of an organizational user (e.g., VPN, proxy capability). This does not apply to authentication for the purpose of configuring the device itself (management).

Satisfies: SRG-OS-000375-GPOS-00160, SRG-OS-000105-GPOS-00052, SRG-OS-000106-GPOS-00053, SRG-OS-000107-GPOS-00054, SRG-OS-000108-GPOS-00055

Rule Check

Verify Ubuntu 22.04 LTS has the packages required for multifactor authentication installed by using the following command:

```
$ dpkg -l | grep libpam-pkcs11
ii     libpam-pkcs11     0.6.11-4build2     amd64     Fully featured PAM module for using PKCS#11 smart cards
```

If the "libpam-pkcs11" package is not installed, this is a finding.

Fix

Install the "libpam-pkcs11" package by using the following command:

```
$ sudo apt-get install libpam-pkcs11
```

V-260574

Title

Ubuntu 22.04 LTS must accept personal identity verification (PIV) credentials.

Description

The use of PIV credentials facilitates standardization and reduces the risk of unauthorized access.

DOD has mandated the use of the common access card (CAC) to support identity management and personal authentication for systems covered under Homeland Security Presidential Directive (HSPD) 12, as well as making the CAC a primary component of layered protection for national security systems.

Rule Check

Verify the "opensc-pcks11" package is installed on the system by using the following command:

```
$ dpkg -l | grep opensc-pkcs11
ii     opensc-pkcs11:amd64     0.22.0-1Ubuntu2     amd64     Smart card utilities with support for PKCS#15 compatible cards
```

If the "opensc-pcks11" package is not installed, this is a finding.

Fix

Install the "opensc-pkcs11" package by using the following command:

```
$ sudo apt-get install opensc-pkcs11
```

V-260575

Title

Ubuntu 22.04 LTS must implement smart card logins for multifactor authentication for local and network access to privileged and nonprivileged accounts.

Description

Without the use of multifactor authentication, the ease of access to privileged functions is greatly increased.

Multifactor authentication requires using two or more factors to achieve authentication.

Factors include:
1) Something a user knows (e.g., password/PIN);
2) Something a user has (e.g., cryptographic identification device, token); and
3) Something a user is (e.g., biometric).

A privileged account is defined as an information system account with authorizations of a privileged user.

Network access is defined as access to an information system by a user (or a process acting on behalf of a user) communicating through a network (e.g., local area network, wide area network, or the internet).

The DOD common access card (CAC) with DOD-approved PKI is an example of multifactor authentication.

Satisfies: SRG-OS-000105-GPOS-00052, SRG-OS-000106-GPOS-00053, SRG-OS-000107-GPOS-00054, SRG-OS-000108-GPOS-00055

Rule Check

Verify that the "pam_pkcs11.so" module is configured by using the following command:

```
$ grep -i pam_pkcs11.so /etc/pam.d/common-auth
auth     [success=2 default=ignore]     pam_pkcs11.so
```

If "pam_pkcs11.so" is commented out, or is missing, this is a finding.

Verify the sshd daemon allows public key authentication by using the following command:

```
$ sudo /usr/sbin/sshd -dd 2>&1 | awk '/filename/ {print $4}' | tr -d '\r' | tr '\n' ' ' | xargs sudo grep -iH 'pubkeyauthentication'
/etc/ssh/sshd_config:PubkeyAuthentication yes
```

If "PubkeyAuthentication" is not set to "yes", is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to use multifactor authentication for access to accounts.

Add or modify the following line in the "/etc/pam.d/common-auth" file:

auth [success=2 default=ignore] pam_pkcs11.so

Add or modify the following line in the "/etc/ssh/sshd_config" file:

PubkeyAuthentication yes

V-260576

Title

Ubuntu 22.04 LTS must electronically verify personal identity verification (PIV) credentials.

Description

The use of PIV credentials facilitates standardization and reduces the risk of unauthorized access.

DOD has mandated the use of the common access card (CAC) to support identity management and personal authentication for systems covered under Homeland Security Presidential Directive (HSPD) 12, as well as making the CAC a primary component of layered protection for national security systems.

Rule Check

Verify Ubuntu 22.04 LTS electronically verifies PIV credentials via certificate status checking by using the following command:

```
$ sudo grep use_pkcs11_module /etc/pam_pkcs11/pam_pkcs11.conf | awk '/pkcs11_module opensc {/,/}/' /etc/pam_pkcs11/pam_pkcs11.conf | grep cert_policy | grep ocsp_on
cert_policy = ca,signature,ocsp_on;
```

If every returned "cert_policy" line is not set to "ocsp_on", the line is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to do certificate status checking for multifactor authentication.

Add or modify all "cert_policy" lines in the "/etc/pam_pkcs11/pam_pkcs11.conf" file with the following:

ocsp_on

V-260577

Title

Ubuntu 22.04 LTS, for PKI-based authentication, must validate certificates by constructing a certification path (which includes status information) to an accepted trust anchor.

Description

Without path validation, an informed trust decision by the relying party cannot be made when presented with any certificate not already explicitly trusted.

A trust anchor is an authoritative entity represented via a public key and associated data. It is used in the context of public key infrastructures, X.509 digital certificates, and DNSSEC.

When there is a chain of trust, usually the top entity to be trusted becomes the trust anchor; it can be, for example, a certification authority (CA). A certification path starts with the subject certificate and proceeds through a number of intermediate certificates up to a trusted root certificate, typically issued by a trusted CA.

This requirement verifies that a certification path to an accepted trust anchor is used for certificate validation and that the path includes status information. Path validation is necessary for a relying party to make an informed trust decision when presented with any certificate not already explicitly trusted. Status information for certification paths includes certificate revocation lists or online certificate status protocol responses. Validation of the certificate status information is out of scope for this requirement.

Rule Check

Verify Ubuntu 22.04 LTS, for PKI-based authentication, has valid certificates by constructing a certification path to an accepted trust anchor.

Determine which pkcs11 module is being used via the "use_pkcs11_module" in "/etc/pam_pkcs11/pam_pkcs11.conf" and then ensure "ca" is enabled in "cert_policy" by using the following command:

```
$ sudo grep use_pkcs11_module /etc/pam_pkcs11/pam_pkcs11.conf | awk '/pkcs11_module opensc {/,/}/' /etc/pam_pkcs11/pam_pkcs11.conf | grep cert_policy | grep ca
cert_policy = ca,signature,ocsp_on;
```

If "cert_policy" is not set to "ca", the line is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS, for PKI-based authentication, to validate certificates by constructing a certification path to an accepted trust anchor.

Add or modify all "cert_policy" lines in the "/etc/pam_pkcs11/pam_pkcs11.conf" file with the following:

cert_policy = ca,signature,ocsp_on;

Note: If the system is missing an "/etc/pam_pkcs11/" directory and an "/etc/pam_pkcs11/pam_pkcs11.conf", find an example to copy into place and modify accordingly at "/usr/share/doc/libpam-pkcs11/examples/pam_pkcs11.conf.example.gz".

V-260578

Title

Ubuntu 22.04 LTS for PKI-based authentication, must implement a local cache of revocation data in case of the inability to access revocation information via the network.

Description

Without configuring a local cache of revocation data, there is the potential to allow access to users who are no longer authorized (users with revoked certificates).

Rule Check

Verify Ubuntu 22.04 LTS, for PKI-based authentication, uses local revocation data when unable to access it from the network by using the following command:

Note: If smart card authentication is not being used on the system, this is not applicable.

```
$ grep cert_policy /etc/pam_pkcs11/pam_pkcs11.conf | grep  -E -- 'crl_auto|crl_offline'
cert_policy = ca,signature,ocsp_on,crl_auto;
```

If "cert_policy" is not set to include "crl_auto" or "crl_offline", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS, for PKI-based authentication, to use local revocation data when unable to access the network to obtain it remotely.

Add or update the "cert_policy" option in "/etc/pam_pkcs11/pam_pkcs11.conf" to include "crl_auto" or "crl_offline".

cert_policy = ca,signature,ocsp_on, crl_auto;

If the system is missing an "/etc/pam_pkcs11/" directory and an "/etc/pam_pkcs11/pam_pkcs11.conf", find an example to copy into place and modify accordingly at "/usr/share/doc/libpam-pkcs11/examples/pam_pkcs11.conf.example.gz".

V-260579

Title

Ubuntu 22.04 LTS must map the authenticated identity to the user or group account for PKI-based authentication.

Description

Without mapping the certificate used to authenticate to the user account, the ability to determine the identity of the individual user or group will not be available for forensic analysis.

Rule Check

Verify that "use_mappers" is set to "pwent" in "/etc/pam_pkcs11/pam_pkcs11.conf" file by using the following command:

```
$ grep -i use_mappers /etc/pam_pkcs11/pam_pkcs11.conf
use_mappers = pwent
```

If "use_mappers" does not contain "pwent", is commented out, or is missing, this is a finding.

Fix

Set "use_mappers=pwent" in "/etc/pam_pkcs11/pam_pkcs11.conf" or, if there is already a comma-separated list of mappers, add it to the list, separated by comma, and before the null mapper.

If the system is missing an "/etc/pam_pkcs11/" directory and an "/etc/pam_pkcs11/pam_pkcs11.conf", find an example to copy into place and modify accordingly at "/usr/share/doc/libpam-pkcs11/examples/pam_pkcs11.conf.example.gz".

V-260580

Title

Ubuntu 22.04 LTS must use DOD PKI-established certificate authorities for verification of the establishment of protected sessions.

Description

Untrusted certificate authorities (CA) can issue certificates, but they may be issued by organizations or individuals that seek to compromise DOD systems or by organizations with insufficient security controls. If the CA used for verifying the certificate is not a DOD-approved CA, trust of this CA has not been established.

The DOD will only accept PKI-certificates obtained from a DOD-approved internal or external certificate authority. Reliance on CAs for the establishment of secure sessions includes, for example, the use of SSL/TLS certificates.

Rule Check

Verify the directory containing the root certificates for Ubuntu 22.04 LTS contains certificate files for DOD PKI-established certificate authorities by iterating over all files in the "/etc/ssl/certs" directory and checking if, at least one, has the subject matching "DOD ROOT CA".

```
$ ls /etc/ssl/certs | grep -i DOD
DOD_PKE_CA_chain.pem
```

If no DOD root certificate is found, this is a finding.

Verify that all root certificates present on the system have been approved by the AO.

```
$ ls /etc/ssl/certs
```

If a certificate is present that is not approved by the AO, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to use of DOD PKI-established certificate authorities for verification of the establishment of protected sessions.

Add at least one DOD certificate authority to the "/usr/share/ca-certificates" directory in the CRT format.

Update the "/etc/ssl/certs" directory by using the following command:

```
$ sudo dpkg-reconfigure ca-certificates
```

V-260581

Title

Ubuntu 22.04 LTS must be configured such that Pluggable Authentication Module (PAM) prohibits the use of cached authentications after one day.

Description

If cached authentication information is out-of-date, the validity of the authentication information may be questionable.

Rule Check

Verify that PAM prohibits the use of cached authentications after one day by using the following command:

Note: If smart card authentication is not being used on the system, this requirement is not applicable.

```
$ sudo grep -i '^\s*offline_credentials_expiration' /etc/sssd/sssd.conf /etc/sssd/conf.d/*.conf
/etc/sssd/sssd.conf:offline_credentials_expiration = 1
```

If "offline_credentials_expiration" is not set to "1", is commented out, is missing, or conflicting results are returned, this is a finding.

Fix

Configure PAM to prohibit the use of cached authentications after one day.

Add or modify the following line in the "/etc/sssd/sssd.conf" file, just below the line "[pam]":

offline_credentials_expiration = 1

Note: It is valid for this configuration to be in a file with a name that ends with ".conf" and does not begin with a "." in the "/etc/sssd/conf.d/" directory instead of the "/etc/sssd/sssd.conf" file.

V-260582

Title

Ubuntu 22.04 LTS must use a file integrity tool to verify correct operation of all security functions.

Description

Without verification of the security functions, security functions may not operate correctly and the failure may go unnoticed. Security function is defined as the hardware, software, and/or firmware of the information system responsible for enforcing the system security policy and supporting the isolation of code and data on which the protection is based. Security functionality includes, but is not limited to, establishing system accounts, configuring access authorizations (i.e., permissions, privileges), setting events to be audited, and setting intrusion detection parameters.

This requirement applies to Ubuntu 22.04 LTS performing security function verification/testing and/or systems and environments that require this functionality.

Rule Check

Verify that Advanced Intrusion Detection Environment (AIDE) is installed by using the following command:

```
$ dpkg -l | grep aide
ii     aide     0.17.4-1     amd64     Advanced Intrusion Detection Environment - dynamic binary
```

If AIDE is not installed, ask the system administrator how file integrity checks are performed on the system.

If there is no application installed to perform integrity checks, this is a finding.

Fix

Install the "aide" package:

```
$ sudo apt install aide
```

V-260583

Title

Ubuntu 22.04 LTS must configure AIDE to perform file integrity checking on the file system.

Description

Without verification of the security functions, security functions may not operate correctly and the failure may go unnoticed. Security function is defined as the hardware, software, and/or firmware of the information system responsible for enforcing the system security policy and supporting the isolation of code and data on which the protection is based. Security functionality includes, but is not limited to, establishing system accounts, configuring access authorizations (i.e., permissions, privileges), setting events to be audited, and setting intrusion detection parameters.

This requirement applies to Ubuntu 22.04 LTS performing security function verification/testing and/or systems and environments that require this functionality.

Rule Check

Verify that Advanced Intrusion Detection Environment (AIDE) is configured and operating correctly by using the following command (this will take a few minutes):

Note: If AIDE is not installed, this requirement is not applicable.

```
$ sudo aide -c /etc/aide/aide.conf --check
```

Example output:

Start timestamp: 2024-04-01 04:20:00 +1300 (AIDE 0.17.4) AIDE found differences between database and filesystem!! Ignored e2fs attributes: EIh ...

If AIDE is being used to perform file integrity checks but the command fails, this is a finding.

Fix

Initialize AIDE (this will take a few minutes):

```
$ sudo aideinit
Running aide --init...
```

Example output:

Start timestamp: 2024-04-01 04:20:00 +1300 (AIDE 0.17.4) AIDE initialized database at /var/lib/aide/aide.db.new Ignored e2fs attributes: EIh

Number of entries: 146185

The attributes of the (uncompressed) database(s):

/var/lib/aide/aide.db.new SHA256 : UrYbC/KBOJcs8zKcSlKoifnnoPK66DEC Aw6odu/BpgY= SHA512 : ezENbbuh937SPWvtsdjRzy3i47XjLg7j L3UGmr0EcgY6u8rczxgbn2RuwJfrIpef 0c1qMNobzrLXyDnnqEqAqw== RMD160 : yBq2xio+g5ne4kvZzzMZ2v+EO9w= TIGER : GkJ/xkzJGu/aSQqk9A5LN271IOAQC3d0 CRC32 : g/beXA== HAVAL : zZm220YZiGna2edJ6Gi0rPv16AlpqeHB y/XLB3hIPEY= WHIRLPOOL : k6veoXavJ/BH9L125pCYAfTB8w5ZJkdC DvVmYS0+cgmg7M0y/S2v42FNCEJ993mc 3kZMXJR/VVmwKg/7ntGixQ== GOST : psjiyix6mJlNsE984D0NwbfgBmB0ETGl /R4PNvm/wKg=

End timestamp: 2024-04-01 04:29:16 +1300 (run time: 9m 16s)

V-260584

Title

Ubuntu 22.04 LTS must notify designated personnel if baseline configurations are changed in an unauthorized manner. The file integrity tool must notify the system administrator when changes to the baseline configuration or anomalies in the operation of any security functions are discovered.

Description

Unauthorized changes to the baseline configuration could make the system vulnerable to various attacks or allow unauthorized access to the operating system. Changes to operating system configurations can have unintended side effects, some of which may be relevant to security.

Detecting such changes and providing an automated response can help avoid unintended, negative consequences that could ultimately affect the security state of the operating system. The operating system's IMO/ISSO and SAs must be notified via email and/or monitoring system trap when there is an unauthorized modification of a configuration item.

Satisfies: SRG-OS-000363-GPOS-00150, SRG-OS-000447-GPOS-00201

Rule Check

Verify that Advanced Intrusion Detection Environment (AIDE) notifies the system administrator when anomalies in the operation of any security functions are discovered by using the following command:

```
$ grep -i '^\s*silentreports' /etc/default/aide
SILENTREPORTS=no
```

If "SILENTREPORTS" is set to "yes", is commented out, or is missing, this is a finding.

Fix

Configure AIDE to notify designated personnel if baseline configurations are changed in an unauthorized manner.

Add or modify the following line in the "/etc/default/aide" file:

SILENTREPORTS=no

V-260585

Title

Ubuntu 22.04 LTS must be configured so that the script that runs each 30 days or less to check file integrity is the default.

Description

Without verification of the security functions, security functions may not operate correctly and the failure may go unnoticed. Security function is defined as the hardware, software, and/or firmware of the information system responsible for enforcing the system security policy and supporting the isolation of code and data on which the protection is based. Security functionality includes, but is not limited to, establishing system accounts, configuring access authorizations (i.e., permissions, privileges), setting events to be audited, and setting intrusion detection parameters.

Notifications provided by information systems include, for example, electronic alerts to system administrators, messages to local computer consoles, and/or hardware indications, such as lights.

This requirement applies to Ubuntu 22.04 LTS performing security function verification/testing and/or systems and environments that require this functionality.

Rule Check

Verify that the Advanced Intrusion Detection Environment (AIDE) default script used to check file integrity each 30 days or less is unchanged.

Download the original aide-common package in the /tmp directory:

```
$ cd /tmp; apt download aide-common
```

Fetch the SHA1 of the original script file:

```
$ dpkg-deb --fsys-tarfile /tmp/aide-common_*.deb | tar -xO ./usr/share/aide/config/cron.daily/aide | shalsum
b71bb2cafaedf15ec3ac2f566f209d3260a37af0  -
```

Compare with the SHA1 of the file in the daily or monthly cron directory:

```
$ shalsum /etc/cron.{daily,monthly}/aide 2>/dev/null
b71bb2cafaedf15ec3ac2f566f209d3260a37af0  /etc/cron.daily/aide
```

If there is no AIDE script file in the cron directories, or the SHA1 value of at least one file in the daily or monthly cron directory does not match the SHA1 of the original, this is a finding.

Fix

The cron file for AIDE is fairly complex as it creates the report. This file is installed with the "aide-common" package, and the default can be restored by copying it from the package:

Extract the aide script from the "aide-common" package to its original place:

```
$ dpkg-deb --fsys-tarfile /tmp/aide-common_*.deb | sudo tar -x ./usr/share/aide/config/cron.daily/aide -C /
```

Copy it to the cron.daily directory:

```
$  sudo cp -f /usr/share/aide/config/cron.daily/aide /etc/cron.daily/aide
```

V-260586

Title

Ubuntu 22.04 LTS must use cryptographic mechanisms to protect the integrity of audit tools.

Description

Protecting the integrity of the tools used for auditing purposes is a critical step toward ensuring the integrity of audit information. Audit information includes all information (e.g., audit records, audit settings, and audit reports) needed to successfully audit information system activity.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

It is not uncommon for attackers to replace the audit tools or inject code into the existing tools with the purpose of providing the capability to hide or erase system activity from the audit logs.

To address this risk, audit tools must be cryptographically signed in order to provide the capability to identify when the audit tools have been modified, manipulated, or replaced. An example is a checksum hash of the file or files.

Rule Check

Verify that Advanced Intrusion Detection Environment (AIDE) is properly configured to use cryptographic mechanisms to protect the integrity of audit tools by using the following command:

```
$ grep -E '(\/sbin\/(audit|au))' /etc/aide/aide.conf
/sbin/auditctl p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/auditd p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/ausearch p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/aureport p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/autrace p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/audispd p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/augenrules p+i+n+u+g+s+b+acl+xattrs+sha512
```

If any of the seven lines do not appear as shown, are commented out, or are missing, this is a finding.

Fix

Configure AIDE to protect the integrity of audit tools:

Add or modify the following lines in the "/etc/aide/aide.conf" file:

Audit Tools

/sbin/auditctl p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/auditd p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/ausearch p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/aureport p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/autrace p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/audispd p+i+n+u+g+s+b+acl+xattrs+sha512
/sbin/augenrules p+i+n+u+g+s+b+acl+xattrs+sha512

V-260587

Title

Ubuntu 22.04 LTS must have a crontab script running weekly to offload audit events of standalone systems.

Description

Information stored in one location is vulnerable to accidental or incidental deletion or alteration.

Offloading is a common process in information systems with limited audit storage capacity.

Rule Check

Verify there is a script that offloads audit data and that script runs weekly by using the following command:

Note: If the system is not connected to a network, this requirement is not applicable.

```
$ ls /etc/cron.weekly
<audit_offload_script_name>
```

Check if the script inside the file does offloading of audit logs to external media.

If the script file does not exist or does not offload audit logs, this is a finding.

Fix

Create a script that offloads audit logs to external media and runs weekly.

The script must be located in the "/etc/cron.weekly" directory.

V-260588

Title

Ubuntu 22.04 LTS must be configured to preserve log records from failure events.

Description

Failure to a known state can address safety or security in accordance with the mission/business needs of the organization. Failure to a known secure state helps prevent a loss of confidentiality, integrity, or availability in the event of a failure of the information system or a component of the system.

Preserving operating system state information helps to facilitate operating system restart and return to the operational mode of the organization with least disruption to mission/business processes.

Rule Check

Verify the log service is installed properly by using the following command:

```
$ dpkg -l | grep rsyslog
ii     rsyslog     8.2112.0-2ubuntu2.2     amd64     reliable system and kernel logging daemon
```

If the "rsyslog" package is not installed, this is a finding.

Check that the log service is enabled and active by using the following commands:

```
$ systemctl is-enabled rsyslog.service
enabled

$ systemctl is-active rsyslog.service
active
```

If "rsyslog.service" is not enabled and active, this is a finding.

Fix

Install the log service by using the following command:

```
$ sudo apt-get install rsyslog
```

Enable and activate the log service by using the following command:

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

V-260589

Title

Ubuntu 22.04 LTS must monitor remote access methods.

Description

Remote access services, such as those providing remote access to network devices and information systems, which lack automated monitoring capabilities, increase risk and make remote user access management difficult at best.

Remote access is access to DOD nonpublic information systems by an authorized user (or an information system) communicating through an external, nonorganization-controlled network. Remote access methods include, for example, dial-up, broadband, and wireless.

Automated monitoring of remote access sessions allows organizations to detect cyberattacks and also ensure ongoing compliance with remote access policies by auditing connection activities of remote access capabilities, such as Remote Desktop Protocol (RDP), on a variety of information system components (e.g., servers, workstations, notebook computers, smartphones, and tablets).

Rule Check

Verify that Ubuntu 22.04 LTS monitors all remote access methods by using the following command:

```
$  grep -Er '^(auth\.\*,authpriv\.\*|daemon\.\*)' /etc/rsyslog.*
/etc/rsyslog.d/50-default.conf:auth.*,authpriv.* /var/log/secure
/etc/rsyslog.d/50-default.conf:daemon.* /var/log/messages
```

If "auth.*", "authpriv.*", or "daemon.*" are not configured to be logged in at least one of the config files, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to monitor all remote access methods.

Add or modify the following line in the "/etc/rsyslog.d/50-default.conf" file:

auth.,*authpriv.* /var/log/secure
daemon.* /var/log/messages

Restart "rsyslog.service" for the changes to take effect by using the following command:

```
$ sudo systemctl restart rsyslog.service
```

V-260590

Title

Ubuntu 22.04 LTS must have the "auditd" package installed.

Description

Without establishing the when, where, type, source, and outcome of events that occurred, it would be difficult to establish, correlate, and investigate the events leading up to an outage or attack.

Without the capability to generate audit records, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit record content that may be necessary to satisfy this requirement includes, for example, time stamps, source and destination addresses, user/process identifiers, event descriptions, success/fail indications, filenames involved, and access control or flow control rules invoked.

Reconstruction of harmful events or forensic analysis is not possible if audit records do not contain enough information.

Successful incident response and auditing relies on timely, accurate system information and analysis in order to allow the organization to identify and respond to potential incidents in a proficient manner. If the operating system does not provide the ability to centrally review the operating system logs, forensic analysis is negatively impacted.

Associating event types with detected events in Ubuntu 22.04 LTS audit logs provides a means of investigating an attack; recognizing resource utilization or capacity thresholds; or identifying an improperly configured operating system.

Satisfies: SRG-OS-000037-GPOS-00015, SRG-OS-000038-GPOS-00016, SRG-OS-000039-GPOS-00017, SRG-OS-000040-GPOS-00018, SRG-OS-000041-GPOS-00019, SRG-OS-000042-GPOS-00020, SRG-OS-000042-GPOS-00021, SRG-OS-000051-GPOS-00024, SRG-OS-000054-GPOS-00025, SRG-OS-000062-GPOS-00031, SRG-OS-000122-GPOS-00063, SRG-OS-000337-GPOS-00129, SRG-OS-000348-GPOS-00136, SRG-OS-000349-GPOS-00137, SRG-OS-000350-GPOS-00138, SRG-OS-000351-GPOS-00139, SRG-OS-000352-GPOS-00140, SRG-OS-000353-GPOS-00141, SRG-OS-000354-GPOS-00142, SRG-OS-000365-GPOS-00152, SRG-OS-000475-GPOS-00220

Rule Check

Verify the "auditd" package is installed by using the following command:

```
$ dpkg -l | grep auditd
ii     libauditd     1:3.0.7-1build1     amd64     User space tools for security auditing
```

If the "auditd" package is not installed,Â this is a finding.

Fix

Install the "auditd" package by using the following command:

```
$ sudo apt-get install auditd
```

V-260591

Title

Ubuntu 22.04 LTS must produce audit records and reports containing information to establish when, where, what type, the source, and the outcome for all DOD-defined auditable events and actions in near real time.

Description

Without establishing the when, where, type, source, and outcome of events that occurred, it would be difficult to establish, correlate, and investigate the events leading up to an outage or attack.

Without the capability to generate audit records, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit record content that may be necessary to satisfy this requirement includes, for example, time stamps, source and destination addresses, user/process identifiers, event descriptions, success/fail indications, filenames involved, and access control or flow control rules invoked.

Reconstruction of harmful events or forensic analysis is not possible if audit records do not contain enough information.

Successful incident response and auditing relies on timely, accurate system information and analysis to allow the organization to identify and respond to potential incidents in a proficient manner. If the operating system does not provide the ability to centrally review the operating system logs, forensic analysis is negatively impacted.

Associating event types with detected events in Ubuntu 22.04 LTS audit logs provides a means of investigating an attack; recognizing resource utilization or capacity thresholds; or identifying an improperly configured operating system.

Satisfies: SRG-OS-000037-GPOS-00015, SRG-OS-000038-GPOS-00016, SRG-OS-000039-GPOS-00017, SRG-OS-000040-GPOS-00018, SRG-OS-000041-GPOS-00019, SRG-OS-000042-GPOS-00020, SRG-OS-000042-GPOS-00021, SRG-OS-000051-GPOS-00024, SRG-OS-000054-GPOS-00025, SRG-OS-000062-GPOS-00031, SRG-OS-000122-GPOS-00063, SRG-OS-000337-GPOS-00129, SRG-OS-000348-GPOS-00136, SRG-OS-000349-GPOS-00137, SRG-OS-000350-GPOS-00138, SRG-OS-000351-GPOS-00139, SRG-OS-000352-GPOS-00140, SRG-OS-000353-GPOS-00141, SRG-OS-000354-GPOS-00142, SRG-OS-000365-GPOS-00152, SRG-OS-000475-GPOS-00220

Rule Check

Verify the "auditd.service" is enabled and active by using the following commands:

```
$ systemctl is-enabled auditd.service
enabled

$ systemctl is-active auditd.service
active
```

If the "auditd.service" is not enabled and active, this is a finding.

Fix

Enable and start the "auditd.service" by using the following command:

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

V-260592

Title

Ubuntu 22.04 LTS audit event multiplexor must be configured to offload audit logs onto a different system from the system being audited.

Description

Information stored in one location is vulnerable to accidental or incidental deletion or alteration.

Offloading is a common process in information systems with limited audit storage capacity.

The auditd service does not include the ability to send audit records to a centralized server for management directly. However, it can use a plug-in for audit event multiplexor to pass audit records to a remote server.

Satisfies: SRG-OS-000342-GPOS-00133, SRG-OS-000479-GPOS-00224

Rule Check

Verify the audit event multiplexor is configured to offload audit records to a different system from the system being audited.

Check if the "audispd-plugins" package is installed:

```
$ dpkg -l | grep audispd-plugins
ii     audispd-plugins     1:3.0.7-1build1     amd64     Plugins for the audit event dispatcher
```

If the "audispd-plugins" package is not installed, this is a finding.

Check that the records are being offloaded to a remote server by using the following command:

```
$ sudo grep -i active /etc/audit/plugins.d/au-remote.conf
active = yes
```

If "active" is not set to "yes", or the line is commented out, or is missing, this is a finding.

Check that audisp-remote plugin is configured to send audit logs to a different system:

```
$ sudo grep -i remote_server /etc/audit/audisp-remote.conf
remote_server = 240.9.19.81
```

If the "remote_server" parameter is not set, is set with a local IP address, or is set with an invalid IP address, this is a finding.

Fix

Configure the audit event multiplexor to offload audit records to a different system from the system being audited.

Install the "audisp-plugins" package by using the following command:

```
$ sudo apt-get install audispd-plugins
```

Set the audisp-remote plugin as active by editing the "/etc/audit/plugins.d/au-remote.conf" file:

```
$ sudo sed -i -E 's/active\s*=\s*no/active = yes/' /etc/audit/plugins.d/au-remote.conf
```

Set the IP address of the remote system by editing the "/etc/audit/audisp-remote.conf" file:

```
$ sudo sed -i -E 's/(remote_server\s*=).*/\1 <remote_server_ip_address>/' /etc/audit/audisp-remote.conf
```

Restart the "auditd.service" for the changes to take effect:

```
$ sudo systemctl restart auditd.service
```

V-260593

Title

Ubuntu 22.04 LTS must alert the information system security officer (ISSO) and system administrator (SA) in the event of an audit processing failure.

Description

It is critical for the appropriate personnel to be aware if a system is at risk of failing to process audit logs as required. Without this notification, the security personnel may be unaware of an impending failure of the audit capability, and system operation may be adversely affected.

Audit processing failures include software/hardware errors, failures in the audit capturing mechanisms, and audit storage capacity being reached or exceeded.

This requirement applies to each audit data storage repository (i.e., distinct information system component where audit records are stored), the centralized audit storage capacity of organizations (i.e., all audit data storage repositories combined), or both.

Rule Check

Verify that the SA and ISSO are notified in the event of an audit processing failure by using the following command:

Note: An email package must be installed on the system for email notifications to be sent.

```
$ sudo grep -i action_mail_acct /etc/audit/auditd.conf
action_mail_acct = <administrator_email_account>
```

If "action_mail_acct" is not set to the email address of the SA and/or ISSO, is commented out, or is missing, this is a finding.

Fix

Configure "auditd" service to notify the SA and ISSO in the event of an audit processing failure.

Add or modify the following line in the "/etc/audit/auditd.conf " file:

action_mail_acct =

Note: Change "administrator_email_account" to the email address of the SA and/or ISSO.

Restart the "auditd" service for the changes take effect:

```
$ sudo systemctl restart auditd.service
```

V-260594

Title

Ubuntu 22.04 LTS must shut down by default upon audit failure.

Description

It is critical that when the operating system is at risk of failing to process audit logs as required, it takes action to mitigate the failure. Audit processing failures include: software/hardware errors; failures in the audit capturing mechanisms; and audit storage capacity being reached or exceeded. Responses to audit failure depend upon the nature of the failure mode.

When availability is an overriding concern, other approved actions in response to an audit failure are as follows:

1. If the failure was caused by the lack of audit record storage capacity, the operating system must continue generating audit records if possible (automatically restarting the audit service if necessary), overwriting the oldest audit records in a first-in-first-out manner.
2. If audit records are sent to a centralized collection server and communication with this server is lost or the server fails, the operating system must queue audit records locally until communication is restored or until the audit records are retrieved manually. Upon restoration of the connection to the centralized collection server, action should be taken to synchronize the local audit data with the collection server.

Rule Check

Verify Ubuntu 22.04 LTS takes the appropriate action when the audit storage volume is full by using the following command:

```
$ sudo grep -i disk_full_action /etc/audit/auditd.conf
disk_full_action = HALT
```

If "disk_full_action" is not set to "HALT", "SYSLOG", or "SINGLE", is commented out, or is missing, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to shut down by default upon audit failure.

Add or modify the following line in the "/etc/audit/auditd.conf " file:

disk_full_action = HALT

Restart the "auditd" service for the changes to take effect:

```
$ sudo systemctl restart auditd.service
```

Note: If system availability has been determined to be more important, and this decision is documented with the ISSO, configure Ubuntu 22.04 LTS to notify system administration staff and ISSO staff in the event of an audit processing failure by setting the "disk_full_action" to "SYSLOG" or "SINGLE".

V-260595

Title

Ubuntu 22.04 LTS must allocate audit record storage capacity to store at least one weeks' worth of audit records, when audit records are not immediately sent to a central audit record storage facility.

Description

To ensure operating systems have a sufficient storage capacity in which to write the audit logs, operating systems must be able to allocate audit record storage capacity.

The task of allocating audit record storage capacity is usually performed during initial installation of the operating system.

Rule Check

Verify Ubuntu 22.04 LTS allocates audit record storage capacity to store at least one week's worth of audit records when audit records are not immediately sent to a central audit record storage facility.

Determine which partition the audit records are being written to by using the following command:

```
$ sudo grep -i log_file /etc/audit/auditd.conf
log_file = /var/log/audit/audit.log
```

Check the size of the partition that audit records are written to (with the example being "/var/log/audit/") by using the following command:

```
$ sudo df -h /var/log/audit/
/dev/sda2 24G 10.4G 13.6G 43% /var/log/audit
```

If the audit records are not written to a partition made specifically for audit records ("/var/log/audit" as a separate partition), determine the amount of space being used by other files in the partition by using the following command:

```
$ sudo du -sh <audit_partition>
1.8G /var/log/audit
```

Note: The partition size needed to capture a week's worth of audit records is based on the activity level of the system and the total storage capacity available.

If the audit record partition is not allocated for sufficient storage capacity, this is a finding.

Fix

Allocate enough storage capacity for at least one week's worth of audit records when audit records are not immediately sent to a central audit record storage facility.

If audit records are stored on a partition made specifically for audit records, use the "parted" program to resize the partition with sufficient space to contain one week's worth of audit records.

If audit records are not stored on a partition made specifically for audit records, a new partition with sufficient amount of space will need be to be created.

Set the auditd server to point to the mount point where the audit records must be located:

```
$ sudo sed -i -E 's@^(log_file\s*=\s*).*@\1 <audit_partition_mountpoint>/audit.log@' /etc/audit/auditd.conf
```

where is the aforementioned mount point.

V-260596

Title

Ubuntu 22.04 LTS must immediately notify the system administrator (SA) and information system security officer (ISSO) when the audit record storage volume reaches 25 percent remaining of the allocated capacity.

Description

If security personnel are not notified immediately when storage volume reaches 25 percent remaining of the allocated capacity, they are unable to plan for audit record storage capacity expansion.

Rule Check

Verify Ubuntu 22.04 LTS is configured to notify the SA and ISSO when the audit record storage volume reaches 25 percent remaining of the allocated capacity by using the following command:

```
$ sudo grep -i space_left /etc/audit/auditd.conf
space_left = 25%
space_left_action = email
```

If "space_left" is set to a value less than "25%", is commented out, or is missing, this is a finding.

If "space_left_action" is not set to "email", is commented out, or is missing, this is a finding.

Note: If the "space_left_action" is set to "exec", the system executes a designated script. If this script informs the SA of the event, this is not a finding.

Fix

Configure Ubuntu 22.04 LTS to notify the SA and ISSO when the audit record storage volume reaches 25 percent remaining of the allocated capacity.

Add or modify the following lines in the "/etc/audit/auditd.conf " file:

space_left = 25% space_left_action = email

Restart the "auditd" service for the changes to take effect:

```
$ sudo systemctl restart auditd.service
```

Note: If the "space_left_action" parameter is set to "exec", ensure the command being executed notifies the SA and ISSO.

V-260597

Title

Ubuntu 22.04 LTS must be configured so that audit log files are not read- or write-accessible by unauthorized users.

Description

Unauthorized disclosure of audit records can reveal system and configuration data to attackers, thus compromising its confidentiality.

Audit information includes all information (e.g., audit records, audit settings, audit reports) needed to successfully audit operating system activity.

Satisfies: SRG-OS-000057-GPOS-00027, SRG-OS-000058-GPOS-00028

Rule Check

Verify that the audit log files have a mode of "600" or less permissive.

Determine where the audit logs are stored by using the following command:

```
$ sudo grep -iw log_file /etc/audit/auditd.conf
log_file = /var/log/audit/audit.log
```

Using the path of the directory containing the audit logs, determine if the audit log files have a mode of "600" or less by using the following command:

```
$ sudo stat -c "%n %a" /var/log/audit/*
/var/log/audit/audit.log 600
```

If the audit log files have a mode more permissive than "600", this is a finding.

Fix

Configure the audit log files to have a mode of "600" or less permissive.

Using the path of the directory containing the audit logs, configure the audit log files to have a mode of "600" or less permissive by using the following command:

```
$ sudo chmod 600 /var/log/audit/*
```

V-260598

Title

Ubuntu 22.04 LTS must be configured to permit only authorized users ownership of the audit log files.

Description

Unauthorized disclosure of audit records can reveal system and configuration data to attackers, thus compromising its confidentiality.

Audit information includes all information (e.g., audit records, audit settings, audit reports) needed to successfully audit operating system activity.

Satisfies: SRG-OS-000057-GPOS-00027, SRG-OS-000058-GPOS-00028, SRG-OS-000059-GPOS-00029

Rule Check

Verify the audit log files are owned by "root" account.

Determine where the audit logs are stored by using the following command:

```
$ sudo grep -iw log_file /etc/audit/auditd.conf
log_file = /var/log/audit/audit.log
```

Using the path of the directory containing the audit logs, determine if the audit log files are owned by the "root" user by using the following command:

```
$ sudo stat -c "%n %U" /var/log/audit/*
/var/log/audit/audit.log root
```

If the audit log files are owned by a user other than "root", this is a finding.

Fix

Configure the audit log directory and its underlying files to be owned by "root" user.

Using the path of the directory containing the audit logs, configure the audit log files to be owned by "root" user by using the following command:

```
$ sudo chown root /var/log/audit/*
```

V-260599

Title

Ubuntu 22.04 LTS must permit only authorized groups ownership of the audit log files.

Description

Unauthorized disclosure of audit records can reveal system and configuration data to attackers, thus compromising its confidentiality.

Audit information includes all information (e.g., audit records, audit settings, audit reports) needed to successfully audit operating system activity.

Satisfies: SRG-OS-000057-GPOS-00027, SRG-OS-000058-GPOS-00028, SRG-OS-000059-GPOS-00029

Rule Check

Verify the group owner of newly created audit logs is "root" by using the following command:

```
$ sudo grep -iw log_group /etc/audit/auditd.conf
log_group = root
```

If "log_group" is not set to "root", this is a finding.

Fix

Configure the group owner of newly created audit logs to be "root".

Add or modify the following lines in the "/etc/audit/auditd.conf " file:

log_group = root

Reload the configuration file of the audit service to update the group ownership of existing files:

```
$ sudo systemctl kill auditd -s SIGHUP
```

V-260600

Title

Ubuntu 22.04 LTS must be configured so that the audit log directory is not write-accessible by unauthorized users.

Description

If audit information were to become compromised, then forensic analysis and discovery of the true source of potentially malicious system activity is impossible to achieve.

To ensure the veracity of audit information, the operating system must protect audit information from unauthorized deletion. This requirement can be achieved through multiple methods, which will depend upon system architecture and design.

Audit information includes all information (e.g., audit records, audit settings, audit reports) needed to successfully audit information system activity.

Rule Check

Verify that the audit log directory has a mode of "750" or less permissive.

Determine where the audit logs are stored by using the following command:

```
$ sudo grep -iw log_file /etc/audit/auditd.conf
log_file = /var/log/audit/audit.log
```

Using the path of the directory containing the audit logs, determine if the directory has a mode of "750" or less by using the following command:

```
$ sudo stat -c "%n %a" /var/log/audit
/var/log/audit 750
```

If the audit log directory has a mode more permissive than "750", this is a finding.

Fix

Configure the audit log directory to have a mode of "750" or less permissive.

Using the path of the directory containing the audit logs, configure the audit log directory to have a mode of "750" or less permissive by using the following command:

```
$ sudo chmod -R  g-w,o-rwx /var/log/audit
```

V-260601

Title

Ubuntu 22.04 LTS must be configured so that audit configuration files are not write-accessible by unauthorized users.

Description

Without the capability to restrict which roles and individuals can select which events are audited, unauthorized personnel may be able to prevent the auditing of critical events.

Misconfigured audits may degrade the system's performance by overwhelming the audit log. Misconfigured audits may also make it more difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Rule Check

Verify that "/etc/audit/audit.rules", "/etc/audit/auditd.conf", and "/etc/audit/rules.d/*" files have a mode of "640" or less permissive by using the following command:

```
$ sudo ls -al /etc/audit/audit.rules /etc/audit/auditd.conf /etc/audit/rules.d/* | awk '{print $1, $9}'
-rw-r----- /etc/audit/audit.rules
-rw-r----- /etc/audit/auditd.conf
-rw-r----- /etc/audit/rules.d/audit.rules
```

If "/etc/audit/audit.rules", "/etc/audit/auditd.conf", or "/etc/audit/rules.d/*" files have a mode more permissive than "640", this is a finding.

Fix

Configure /etc/audit/audit.rules", "/etc/audit/auditd.conf", and "/etc/audit/rules.d/*" files to have a mode of "640" by using the following command:

```
$ sudo chmod -R 640 /etc/audit/audit.rules /etc/audit/auditd.conf /etc/audit/rules.d/*
```

V-260602

Title

Ubuntu 22.04 LTS must permit only authorized accounts to own the audit configuration files.

Description

Without the capability to restrict which roles and individuals can select which events are audited, unauthorized personnel may be able to prevent the auditing of critical events.

Misconfigured audits may degrade the system's performance by overwhelming the audit log. Misconfigured audits may also make it more difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Rule Check

Verify that "/etc/audit/audit.rules", "/etc/audit/auditd.conf", and "/etc/audit/rules.d/*" files are owned by root account by using the following command:

```
$ sudo ls -al /etc/audit/audit.rules /etc/audit/auditd.conf /etc/audit/rules.d/* | awk '{print $3, $9}'
root /etc/audit/audit.rules
root /etc/audit/auditd.conf
root /etc/audit/rules.d/audit.rules
```

If "/etc/audit/audit.rules", "/etc/audit/auditd.conf", or "/etc/audit/rules.d/*" files are owned by a user other than "root", this is a finding.

Fix

Configure "/etc/audit/audit.rules", "/etc/audit/rules.d/*", and "/etc/audit/auditd.conf" files to be owned by root by using the following command:

```
$ sudo chown -R root /etc/audit/audit.rules /etc/audit/auditd.conf /etc/audit/rules.d/*
```

V-260603

Title

Ubuntu 22.04 LTS must permit only authorized groups to own the audit configuration files.

Description

Without the capability to restrict which roles and individuals can select which events are audited, unauthorized personnel may be able to prevent the auditing of critical events.

Misconfigured audits may degrade the system's performance by overwhelming the audit log. Misconfigured audits may also make it more difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Rule Check

Verify that "/etc/audit/audit.rules", "/etc/audit/auditd.conf", and "/etc/audit/rules.d/*" files are owned by root group by using the following command:

```
$ sudo ls -al /etc/audit/audit.rules /etc/audit/auditd.conf /etc/audit/rules.d/* | awk '{print $4, $9}'
root /etc/audit/audit.rules
root /etc/audit/auditd.conf
root /etc/audit/rules.d/audit.rules
```

If "/etc/audit/audit.rules", "/etc/audit/auditd.conf", or "/etc/audit/rules.d/*" files are owned by a group other than "root", this is a finding.

Fix

Configure "/etc/audit/audit.rules", "/etc/audit/rules.d/*", and "/etc/audit/auditd.conf" files to be owned by root group by using the following command:

```
$ sudo chown -R :root /etc/audit/audit.rules /etc/audit/auditd.conf /etc/audit/rules.d/*
```

V-260604

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the apparmor_parser command.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "apparmor_parser" command by using the following command:

```
$ sudo auditctl -l | grep apparmor_parser
-a always,exit -S all -F path=/sbin/apparmor_parser -F perm=x -F auid>=1000 -F auid!=-1 -F key=perm_chng
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "apparmor_parser" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/sbin/apparmor_parser -F perm=x -F auid>=1000 -F auid!=unset -k perm_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260605

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chacl command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "chacl" command by using the following command:

```
$ sudo auditctl -l | grep chacl
-a always,exit -S all -F path=/usr/bin/chacl -F perm=x -F auid>=1000 -F auid!=-1 -F key=perm_chng
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "chacl" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/chacl -F perm=x -F auid>=1000 -F auid!=unset -k perm_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260606

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chage command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "chage" command by using the following command:

```
$ sudo auditctl -l | grep -w chage
-a always,exit -S all -F path=/usr/bin/chage -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-chage
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "chage" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/chage -F perm=x -F auid>=1000 -F auid!=unset -k privileged-chage

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260607

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chcon command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "chcon" command by using the following command:

```
$ sudo auditctl -l | grep chcon
-a always,exit -S all -F path=/usr/bin/chcon -F perm=x -F auid>=1000 -F auid!=-1 -F key=perm_chng
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "chcon" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/chcon -F perm=x -F auid>=1000 -F auid!=unset -k perm_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260608

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chfn command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records upon successful/unsuccessful attempts to use the "chfn" command by using the following command:

```
$ sudo auditctl -l | grep /usr/bin/chfn
-a always,exit -S all -F path=/usr/bin/chfn -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-chfn
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "chfn" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/chfn -F perm=x -F auid>=1000 -F auid!=unset -k privileged-chfn

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260609

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chsh command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "chsh" command by using the following command:

```
$ sudo auditctl -l | grep chsh
-a always,exit -S all -F path=/usr/bin/chsh -F perm=x -F auid>=1000 -F auid!=-1 -F key=priv_cmd
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Notes: The "-k" allows for specifying an arbitrary identifier, and the string after it does not need to match the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "chsh" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/chsh -F perm=x -F auid>=1000 -F auid!=unset -k priv_cmd

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260610

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the crontab command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "crontab" command by using the following command:

```
$ sudo auditctl -l | grep -w crontab
-a always,exit -S all -F path=/usr/bin/crontab -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-crontab
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "crontab" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/crontab -F perm=x -F auid>=1000 -F auid!=unset -k privileged-crontab

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260611

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful attempts to use the fdisk command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS is configured to audit the execution of the partition management program "fdisk" by using the following command:

```
$ sudo auditctl -l | grep fdisk
-w /usr/sbin/fdisk -p x -k fdisk
```

If the command does not return a line, or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to audit the execution of the partition management program "fdisk".

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /usr/sbin/fdisk -p x -k fdisk

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260612

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the gpasswd command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "gpasswd" command by using the following command:

```
$ sudo auditctl -l | grep -w gpasswd
-a always,exit -S all -F path=/usr/bin/gpasswd -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-gpasswd
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "gpasswd" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/gpasswd -F perm=x -F auid>=1000 -F auid!=unset -k privileged-gpasswd

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260613

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful attempts to use the kmod command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS is configured to audit the execution of the module management program "kmod" by using the following command:

```
$ sudo auditctl -l | grep kmod
-w /bin/kmod -p x -k module
```

If the command does not return a line, or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to audit the execution of the module management program "kmod".

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /bin/kmod -p x -k modules

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260614

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful attempts to use modprobe command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify if Ubuntu 22.04 LTS is configured to audit the execution of the module management program "modprobe" with the following command:

```
$ sudo auditctl -l | grep /sbin/modprobe
-w /sbin/modprobe -p x -k modules
```

If the command does not return a line, or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to audit the execution of the module management program "modprobe".

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /sbin/modprobe -p x -k modules

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260615

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the mount command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records upon successful/unsuccessful attempts to use the "mount" command by using the following command:

```
$ sudo auditctl -l | grep /usr/bin/mount
-a always,exit -S all -F path=/usr/bin/mount -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-mount
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "mount" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/mount -F perm=x -F auid>=1000 -F auid!=unset -k privileged-mount

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260616

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the newgrp command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "newgrp" command by using the following command:

```
$ sudo auditctl -l | grep newgrp
-a always,exit -S all -F path=/usr/bin/newgrp -F perm=x -F auid>=1000 -F auid!=-1 -F key=priv_cmd
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "newgrp" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/newgrp -F perm=x -F auid>=1000 -F auid!=unset -k priv_cmd

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260617

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the pam_timestamp_check command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "pam_timestamp_check" command by using the following command:

```
$ sudo auditctl -l | grep -w pam_timestamp_check
-a always,exit -S all -F path=/usr/sbin/pam_timestamp_check -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-pam_timestamp_check
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "pam_timestamp_check" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/sbin/pam_timestamp_check -F perm=x -F auid>=1000 -F auid!=unset -k privileged-pam_timestamp_check

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260618

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the passwd command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "passwd" command by using the following command:

```
$ sudo auditctl -l | grep -w passwd
-a always,exit -S all -F path=/usr/bin/passwd -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-passwd
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key" allows for specifying an arbitrary identifier, and the string after it does not need to match the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "passwd" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/passwd -F perm=x -F auid>=1000 -F auid!=unset -k privileged-passwd

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260619

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the setfacl command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "setfacl" command by using the following command:

```
$ sudo auditctl -l | grep setfacl
-a always,exit -S all -F path=/usr/bin/setfacl -F perm=x -F auid>=1000 -F auid!=-1 -F key=perm_chng
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "setfacl" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/setfacl -F perm=x -F auid>=1000 -F auid!=unset -k perm_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260620

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the ssh-agent command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "ssh-agent" command by using the following command:

```
$ sudo auditctl -l | grep /usr/bin/ssh-agent
-a always,exit -S all -F path=/usr/bin/ssh-agent -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-ssh
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "ssh-agent" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/ssh-agent -F perm=x -F auid>=1000 -F auid!=unset -k privileged-ssh

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260621

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the ssh-keysign command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "ssh-keysign" command by using the following command:

```
$ sudo auditctl -l | grep ssh-keysign
-a always,exit -S all -F path=/usr/lib/openssh/ssh-keysign -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-ssh
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "ssh-keysign" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/lib/openssh/ssh-keysign -F perm=x -F auid>=1000 -F auid!=unset -k privileged-ssh

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260622

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the su command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records upon successful/unsuccessful attempts to use the "su" command by using the following command:

```
$ sudo auditctl -l | grep /bin/su
-a always,exit -S all -F path=/bin/su -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-priv_change
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records when successful/unsuccessful attempts to use the "su" command occur.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/bin/su -F perm=x -F auid>=1000 -F auid!=unset -k privileged-priv_change

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260623

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the sudo command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "sudo" command by using the following command:

```
$ sudo auditctl -l | grep /usr/bin/sudo
-a always,exit -S all -F path=/usr/bin/sudo -F perm=x -F auid>=1000 -F auid!=-1 -F key=priv_cmd
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "sudo" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/sudo -F perm=x -F auid>=1000 -F auid!=unset -k priv_cmd

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260624

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the sudoedit command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "sudoedit" command by using the following command:

```
$ sudo auditctl -l | grep /usr/bin/sudoedit
-a always,exit -S all -F path=/usr/bin/sudoedit -F perm=x -F auid>=1000 -F auid!=-1 -F key=priv_cmd
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "sudoedit" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules":

-a always,exit -F path=/usr/bin/sudoedit -F perm=x -F auid>=1000 -F auid!=unset -k priv_cmd

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260625

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the umount command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify if Ubuntu 22.04 LTS generates audit records upon successful/unsuccessful attempts to use the "umount" command by using the following command:

```
$ sudo auditctl -l | grep /usr/bin/umount
-a always,exit -S all -F path=/usr/bin/umount -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-umount
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "umount" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/bin/umount -F perm=x -F auid>=1000 -F auid!=unset -k privileged-umount

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260626

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the unix_update command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "unix_update" command by using the following command:

```
$ sudo auditctl -l | grep -w unix_update
-a always,exit -S all -F path=/sbin/unix_update -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-unix-update
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "unix_update" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/sbin/unix_update -F perm=x -F auid>=1000 -F auid!=unset -k privileged-unix-update

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260627

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the usermod command.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify that an audit event is generated for any successful/unsuccessful use of the "usermod" command by using the following command:

```
$ sudo auditctl -l | grep -w usermod
-a always,exit -S all -F path=/usr/sbin/usermod -F perm=x -F auid>=1000 -F auid!=-1 -F key=privileged-usermod
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful uses of the "usermod" command.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F path=/usr/sbin/usermod -F perm=x -F auid>=1000 -F auid!=unset -k privileged-usermod

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260628

Title

Ubuntu 22.04 LTS must generate audit records for all account creations, modifications, disabling, and termination events that affect /etc/group.

Description

Once an attacker establishes access to a system, the attacker often attempts to create a persistent method of reestablishing access. One way to accomplish this is for the attacker to create an account. Auditing account creation actions provides logging that can be used for forensic purposes.

To address access requirements, many operating systems may be integrated with enterprise level authentication/access/auditing mechanisms that meet or exceed access control policy requirements.

Satisfies: SRG-OS-000004-GPOS-00004, SRG-OS-000239-GPOS-00089, SRG-OS-000240-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000458-GPOS-00203, SRG-OS-000463-GPOS-00207, SRG-OS-000476-GPOS-00221

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all account creations, modifications, disabling, and termination events that affect "/etc/group" by using the following command:

```
$ sudo auditctl -l | grep group
-w /etc/group -p wa -k usergroup_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all account creations, modifications, disabling, and termination events that affect "/etc/group".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/group -p wa -k usergroup_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260629

Title

Ubuntu 22.04 LTS must generate audit records for all account creations, modifications, disabling, and termination events that affect /etc/gshadow.

Description

Once an attacker establishes access to a system, the attacker often attempts to create a persistent method of reestablishing access. One way to accomplish this is for the attacker to create an account. Auditing account creation actions provides logging that can be used for forensic purposes.

To address access requirements, many operating systems may be integrated with enterprise level authentication/access/auditing mechanisms that meet or exceed access control policy requirements.

Satisfies: SRG-OS-000004-GPOS-00004, SRG-OS-000239-GPOS-00089, SRG-OS-000240-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000458-GPOS-00203, SRG-OS-000463-GPOS-00207, SRG-OS-000476-GPOS-00221

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all account creations, modifications, disabling, and termination events that affect "/etc/gshadow" by using the following command:

```
$ sudo auditctl -l | grep gshadow
-w /etc/gshadow -p wa -k usergroup_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all account creations, modifications, disabling, and termination events that affect "/etc/gshadow".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/gshadow -p wa -k usergroup_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260630

Title

Ubuntu 22.04 LTS must generate audit records for all account creations, modifications, disabling, and termination events that affect /etc/opasswd.

Description

Once an attacker establishes access to a system, the attacker often attempts to create a persistent method of reestablishing access. One way to accomplish this is for the attacker to create an account. Auditing account creation actions provides logging that can be used for forensic purposes.

To address access requirements, many operating systems may be integrated with enterprise level authentication/access/auditing mechanisms that meet or exceed access control policy requirements.

Satisfies: SRG-OS-000004-GPOS-00004, SRG-OS-000239-GPOS-00089, SRG-OS-000240-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000458-GPOS-00203, SRG-OS-000463-GPOS-00207, SRG-OS-000476-GPOS-00221

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all account creations, modifications, disabling, and termination events that affect "/etc/security/opasswd" by using the following command:

```
$ sudo auditctl -l | grep opasswd
-w /etc/security/opasswd -p wa -k usergroup_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all account creations, modifications, disabling, and termination events that affect "/etc/security/opasswd".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/security/opasswd -p wa -k usergroup_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260631

Title

Ubuntu 22.04 LTS must generate audit records for all account creations, modifications, disabling, and termination events that affect /etc/passwd.

Description

Once an attacker establishes access to a system, the attacker often attempts to create a persistent method of reestablishing access. One way to accomplish this is for the attacker to create an account. Auditing account creation actions provides logging that can be used for forensic purposes.

To address access requirements, many operating systems may be integrated with enterprise level authentication/access/auditing mechanisms that meet or exceed access control policy requirements.

Satisfies: SRG-OS-000004-GPOS-00004, SRG-OS-000239-GPOS-00089, SRG-OS-000240-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000458-GPOS-00203, SRG-OS-000463-GPOS-00207, SRG-OS-000476-GPOS-00221

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all account creations, modifications, disabling, and termination events that affect "/etc/passwd" by using the following command:

```
$ sudo auditctl -l | grep passwd
-w /etc/passwd -p wa -k usergroup_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all account creations, modifications, disabling, and termination events that affect "/etc/passwd".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/passwd -p wa -k usergroup_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260632

Title

Ubuntu 22.04 LTS must generate audit records for all account creations, modifications, disabling, and termination events that affect /etc/shadow.

Description

Once an attacker establishes access to a system, the attacker often attempts to create a persistent method of reestablishing access. One way to accomplish this is for the attacker to create an account. Auditing account creation actions provides logging that can be used for forensic purposes.

To address access requirements, many operating systems may be integrated with enterprise level authentication/access/auditing mechanisms that meet or exceed access control policy requirements.

Satisfies: SRG-OS-000004-GPOS-00004, SRG-OS-000239-GPOS-00089, SRG-OS-000240-GPOS-00090, SRG-OS-000241-GPOS-00091, SRG-OS-000303-GPOS-00120, SRG-OS-000458-GPOS-00203, SRG-OS-000463-GPOS-00207, SRG-OS-000476-GPOS-00221

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all account creations, modifications, disabling, and termination events that affect "/etc/shadow" by using the following command:

```
$ sudo auditctl -l | grep shadow
-w /etc/shadow -p wa -k usergroup_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all account creations, modifications, disabling, and termination events that affect "/etc/shadow".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/shadow -p wa -k usergroup_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260633

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chmod, fchmod, and fchmodat system calls.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

The system call rules are loaded into a matching engine that intercepts each syscall that all programs on the system makes. Therefore, it is very important to only use syscall rules when absolutely necessary since these affect performance. The more rules, the bigger the performance hit. The performance is helped, though, by combining syscalls into one rule whenever possible.

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000462-GPOS-00206

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "chmod", "fchmod", and "fchmodat" system calls by using the following command:

```
$ sudo auditctl -l | grep chmod
-a always,exit -F arch=b32 -S chmod,fchmod,fchmodat -F auid>=1000 -F auid!=-1 -F key=perm_chng
-a always,exit -F arch=b64 -S chmod,fchmod,fchmodat -F auid>=1000 -F auid!=-1 -F key=perm_chng
```

If the command does not return audit rules for the "chmod", "fchmod" and "fchmodat" syscalls or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "chmod", "fchmod", and "fchmodat" system calls.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules":

-a always,exit -F arch=b32 -S chmod,fchmod,fchmodat -F auid>=1000 -F auid!=unset -k perm_chng -a always,exit -F arch=b64 -S chmod,fchmod,fchmodat -F auid>=1000 -F auid!=unset -k perm_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260634

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the chown, fchown, fchownat, and lchown system calls.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

The system call rules are loaded into a matching engine that intercepts each syscall that all programs on the system makes. Therefore, it is very important to only use syscall rules when absolutely necessary since these affect performance. The more rules, the bigger the performance hit. The performance is helped, though, by combining syscalls into one rule whenever possible.

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000462-GPOS-00206

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "chown", "fchown", "fchownat", and "lchown" system calls by using the following command:

```
$ sudo auditctl -l | grep chown
-a always,exit -F arch=b32 -S chown,fchown,fchownat,lchown -F auid>=1000 -F auid!=-1 -F key=perm_chng
-a always,exit -F arch=b64 -S chown,fchown,fchownat,lchown -F auid>=1000 -F auid!=-1 -F key=perm_chng
```

If the command does not return audit rules for the "chown", "fchown", "fchownat", and "lchown" syscalls or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "chown", "fchown", "fchownat", and "lchown" system calls.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules":

-a always,exit -F arch=b32 -S chown,fchown,fchownat,lchown -F auid>=1000 -F auid!=unset -k perm_chng
-a always,exit -F arch=b64 -S chown,fchown,fchownat,lchown -F auid>=1000 -F auid!=unset -k perm_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260635

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the creat, open, openat, open_by_handle_at, truncate, and ftruncate system calls.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

The system call rules are loaded into a matching engine that intercepts each syscall that all programs on the system makes. Therefore, it is very important to only use syscall rules when absolutely necessary since these affect performance. The more rules, the bigger the performance hit. The performance is helped, though, by combining syscalls into one rule whenever possible.

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000474-GPOS-00219

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon unsuccessful attempts to use the "creat", "open", "openat", "open_by_handle_at", "truncate", and "ftruncate" system calls by using the following command:

```
$ sudo auditctl -l | grep 'open\|truncate\|creat'
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=-1 -F key=perm_access
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=-1 -F key=perm_access
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=-1 -F key=perm_access
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=-1 -F key=perm_access
```

If the command does not return audit rules for the "creat", "open", "openat", "open_by_handle_at", "truncate", and "ftruncate" syscalls or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any unsuccessful use of the "creat", "open", "openat", "open_by_handle_at", "truncate", and "ftruncate" system calls.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -k perm_access
-a always,exit -F arch=b32 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -k perm_access
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EPERM -F auid>=1000 -F auid!=unset -k perm_access
-a always,exit -F arch=b64 -S creat,open,openat,open_by_handle_at,truncate,ftruncate -F exit=-EACCES -F auid>=1000 -F auid!=unset -k perm_access

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260636

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the delete_module system call.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000477-GPOS-00222

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record for any successful/unsuccessful attempts to use the "delete_module" syscall by using the following command:

```
$ sudo auditctl -l | grep -w delete_module
-a always,exit -F arch=b32 -S delete_module -F auid>=1000 -F auid!=-1 -F key=module_chng
-a always,exit -F arch=b64 -S delete_module -F auid>=1000 -F auid!=-1 -F key=module_chng
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "delete_module" syscall.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F arch=b32 -S delete_module -F auid>=1000 -F auid!=unset -k module_chng
-a always,exit -F arch=b64 -S delete_module -F auid>=1000 -F auid!=unset -k module_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260637

Title

Ubuntu 22.04 LTS must generate audit records for successful/unsuccessful uses of the init_module and finit_module system calls.

Description

Without generating audit records that are specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

The system call rules are loaded into a matching engine that intercepts each syscall that all programs on the system makes. Therefore, it is very important to only use syscall rules when absolutely necessary since these affect performance. The more rules, the bigger the performance hit. The performance is helped, though, by combining syscalls into one rule whenever possible.

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000471-GPOS-00216

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record for any successful/unsuccessful attempts to use the "init_module" and "finit_module" syscalls by using the following command:

```
$ sudo auditctl -l | grep init_module
-a always,exit -F arch=b32 -S init_module,finit_module -F auid>=1000 -F auid!=-1 -F key=module_chng
-a always,exit -F arch=b64 -S init_module,finit_module -F auid>=1000 -F auid!=-1 -F key=module_chng
```

If the command does not return audit rules for the "init_module" and "finit_module" syscalls or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "init_module" and "finit_module" syscalls.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F arch=b32 -S init_module,finit_module -F auid>=1000 -F auid!=unset -k module_chng
-a always,exit -F arch=b64 -S init_module,finit_module -F auid>=1000 -F auid!=unset -k module_chng

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260638

Title

Ubuntu 22.04 LTS must generate audit records for any use of the setxattr, fsetxattr, lsetxattr, removexattr, fremovexattr, and lremovexattr system calls.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

The system call rules are loaded into a matching engine that intercepts each syscall that all programs on the system makes. Therefore, it is very important to only use syscall rules when absolutely necessary since these affect performance. The more rules, the bigger the performance hit. The performance is helped, though, by combining syscalls into one rule whenever possible.

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000462-GPOS-00206

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful attempts to use the "setxattr", "fsetxattr", "lsetxattr", "removexattr", "fremovexattr", and "lremovexattr" system calls by using the following command:

```
$ sudo auditctl -l | grep xattr
-a always,exit -F arch=b32 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid>=1000 -F auid!=-1 -F key=perm_mod
-a always,exit -F arch=b32 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid=0 -k perm_mod
-a always,exit -F arch=b64 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid>=1000 -F auid!=-1 -F key=perm_mod
-a always,exit -F arch=b64 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid=0 -k perm_mod
```

If the command does not return audit rules for the "setxattr", "fsetxattr", "lsetxattr", "removexattr", "fremovexattr" and "lremovexattr" syscalls or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful use of the "setxattr", "fsetxattr", "lsetxattr", "removexattr", "fremovexattr", and "lremovexattr" system calls.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F arch=b32 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid>=1000 -F auid!=unset -k perm_mod
-a always,exit -F arch=b32 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid=0 -k perm_mod
-a always,exit -F arch=b64 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid>=1000 -F auid!=unset -k perm_mod
-a always,exit -F arch=b64 -S setxattr,fsetxattr,lsetxattr,removexattr,fremovexattr,lremovexattr -F auid=0 -k perm_mod

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260639

Title

Ubuntu 22.04 LTS must generate audit records for any successful/unsuccessful use of unlink, unlinkat, rename, renameat, and rmdir system calls.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

The system call rules are loaded into a matching engine that intercepts each syscall that all programs on the system makes. Therefore, it is very important to only use syscall rules when absolutely necessary since these affect performance. The more rules, the bigger the performance hit. The performance is helped, though, by combining syscalls into one rule whenever possible.

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for any successful/unsuccessful use of "unlink", "unlinkat", "rename", "renameat", and "rmdir" system calls by using the following command:

```
$ sudo auditctl -l | grep 'unlink\|rename\|rmdir'
-a always,exit -F arch=b64 -S unlink,unlinkat,rename,renameat,rmdir -F auid>=1000 -F auid!=-1 -F key=delete
-a always,exit -F arch=b32 -S unlink,unlinkat,rename,renameat,rmdir -F auid>=1000 -F auid!=-1 -F key=delete
```

If the command does not return audit rules for the "unlink", "unlinkat", "rename", "renameat", and "rmdir" syscalls or the lines are commented out, this is a finding.

Note: The "key" allows for specifying an arbitrary identifier, and the string after it does not need to match the example output above.

Fix

Configure the audit system to generate audit events for any successful/unsuccessful use of "unlink", "unlinkat", "rename", "renameat", and "rmdir" system calls.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F arch=b64 -S unlink,unlinkat,rename,renameat,rmdir -F auid>=1000 -F auid!=unset -k delete
-a always,exit -F arch=b32 -S unlink,unlinkat,rename,renameat,rmdir -F auid>=1000 -F auid!=unset -k delete

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260640

Title

Ubuntu 22.04 LTS must generate audit records for all events that affect the systemd journal files.

Description

Once an attacker establishes access to a system, the attacker often attempts to create a persistent method of reestablishing access. One way to accomplish this is for the attacker to modify system level binaries and their operation. Auditing the systemd journal files provides logging that can be used for forensic purposes.

To address access requirements, many operating systems may be integrated with enterprise level authentication/access/auditing mechanisms that meet or exceed access control policy requirements.

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all events that affect "/var/log/journal" by using the following command:

```
$ sudo auditctl -l | grep journal
-w /var/log/journal -p wa -k systemd_journal
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for events that affect "/var/log/journal".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /var/log/journal -p wa -k systemd_journal

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260641

Title

Ubuntu 22.04 LTS must generate audit records for the /var/log/btmp file.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records showing start and stop times for user access to the system via the "/var/log/btmp" file by using the following command:

```
$ sudo auditctl -l | grep '/var/log/btmp'
-w /var/log/btmp -p wa -k logins
```

If the command does not return a line matching the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate audit events showing start and stop times for user access via the "/var/log/btmp file".

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /var/log/btmp -p wa -k logins

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260642

Title

Ubuntu 22.04 LTS must generate audit records for the /var/log/wtmp file.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records showing start and stop times for user access to the system via the "/var/log/wtmp" file by using the following command:

```
$ sudo auditctl -l | grep '/var/log/wtmp'
-w /var/log/wtmp -p wa -k logins
```

If the command does not return a line matching the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate audit events showing start and stop times for user access via the "/var/log/wtmp" file.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /var/log/wtmp -p wa -k logins

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260643

Title

Ubuntu 22.04 LTS must generate audit records for the /var/run/utmp file.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records showing start and stop times for user access to the system via the "/var/run/utmp" file by using the following command:

```
$ sudo auditctl -l | grep '/var/run/utmp'
-w /var/run/utmp -p wa -k logins
```

If the command does not return a line matching the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate audit events showing start and stop times for user access via the "/var/run/utmp" file.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /var/run/utmp -p wa -k logins

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260644

Title

Ubuntu 22.04 LTS must generate audit records for the use and modification of faillog file.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000470-GPOS-00214, SRG-OS-000473-GPOS-00218

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record upon successful/unsuccessful modifications to the "faillog" file by using the following command:

```
$ sudo auditctl -l | grep faillog
-w /var/log/faillog -p wa -k logins
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful modifications to the "faillog" file.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /var/log/faillog -p wa -k logins

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260645

Title

Ubuntu 22.04 LTS must generate audit records for the use and modification of the lastlog file.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Satisfies: SRG-OS-000064-GPOS-00033, SRG-OS-000470-GPOS-00214, SRG-OS-000473-GPOS-00218

Rule Check

Verify Ubuntu 22.04 LTS generates an audit record when successful/unsuccessful modifications to the "lastlog" file occur by using the following command:

```
$ sudo auditctl -l | grep lastlog
-w /var/log/lastlog -p wa -k logins
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure the audit system to generate an audit event for any successful/unsuccessful modifications to the "lastlog" file.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /var/log/lastlog -p wa -k logins

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260646

Title

Ubuntu 22.04 LTS must generate audit records when successful/unsuccessful attempts to modify the /etc/sudoers file occur.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all modifications that affect "/etc/sudoers" by using the following command:

```
$ sudo auditctl -l | grep sudoers
-w /etc/sudoers -p wa -k privilege_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all modifications that affect "/etc/sudoers".

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/sudoers -p wa -k privilege_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260647

Title

Ubuntu 22.04 LTS must generate audit records when successful/unsuccessful attempts to modify the /etc/sudoers.d directory occur.

Description

Without generating audit records specific to the security and mission needs of the organization, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within the information system (e.g., module or policy filter).

Rule Check

Verify Ubuntu 22.04 LTS generates audit records for all modifications that affect "/etc/sudoers.d" directory by using the following command:

```
$ sudo auditctl -l | grep sudoers.d
-w /etc/sudoers.d -p wa -k privilege_modification
```

If the command does not return a line that matches the example or the line is commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to generate audit records for all modifications that affect "/etc/sudoers.d" directory.

Add or modify the following line to "/etc/audit/rules.d/stig.rules":

-w /etc/sudoers.d -p wa -k privilege_modification

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. he does not need to match the example above.

V-260648

Title

Ubuntu 22.04 LTS must prevent all software from executing at higher privilege levels than users executing the software and the audit system must be configured to audit the execution of privileged functions.

Description

In certain situations, software applications/programs need to execute with elevated privileges to perform required functions. However, if the privileges required for execution are at a higher level than the privileges assigned to organizational users invoking such applications/programs, those users are indirectly provided with greater privileges than assigned by the organizations.

Some programs and processes are required to operate at a higher privilege level and therefore should be excluded from the organization-defined software list after review.

Satisfies: SRG-OS-000326-GPOS-00126, SRG-OS-000327-GPOS-00127

Rule Check

Verify Ubuntu 22.04 LTS audits the execution of privilege functions by auditing the "execve" system call by using the following command:

```
$ sudo auditctl -l | grep execve
-a always,exit -F arch=b64 -S execve -C uid!=euid -F euid=0 -F key=execpriv
-a always,exit -F arch=b64 -S execve -C gid!=egid -F egid=0 -F key=execpriv
-a always,exit -F arch=b32 -S execve -C uid!=euid -F euid=0 -F key=execpriv
-a always,exit -F arch=b32 -S execve -C gid!=egid -F egid=0 -F key=execpriv
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "key=" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to audit the execution of all privileged functions.

Add or modify the following lines in the "/etc/audit/rules.d/stig.rules" file:

-a always,exit -F arch=b64 -S execve -C uid!=euid -F euid=0 -k execpriv -a always,exit -F arch=b64 -S execve -C gid!=egid -F egid=0 -k execpriv -a always,exit -F arch=b32 -S execve -C uid!=euid -F euid=0 -k execpriv -a always,exit -F arch=b32 -S execve -C gid!=egid -F egid=0 -k execpriv

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260649

Title

Ubuntu 22.04 LTS must generate audit records for privileged activities, nonlocal maintenance, diagnostic sessions and other system-level access.

Description

If events associated with nonlocal administrative access or diagnostic sessions are not logged, a major tool for assessing and investigating attacks would not be available.

This requirement addresses auditing-related issues associated with maintenance tools used specifically for diagnostic and repair actions on organizational information systems.

Nonlocal maintenance and diagnostic activities are those activities conducted by individuals communicating through a network, either an external network (e.g., the internet) or an internal network. Local maintenance and diagnostic activities are those activities carried out by individuals physically present at the information system or information system component and not communicating across a network connection.

This requirement applies to hardware/software diagnostic test equipment or tools. This requirement does not cover hardware/software components that may support information system maintenance, yet are a part of the system, for example, the software implementing "ping," "ls," "ipconfig," or the hardware and software implementing the monitoring port of an Ethernet switch.

Satisfies: SRG-OS-000392-GPOS-00172, SRG-OS-000471-GPOS-00215

Rule Check

Verify Ubuntu 22.04 LTS audits activities performed during nonlocal maintenance and diagnostic sessions by using the following command:

```
$ sudo auditctl -l | grep sudo.log
-w /var/log/sudo.log -p wa -k maintenance
```

If the command does not return lines that match the example or the lines are commented out, this is a finding.

Note: The "-k" value is arbitrary and can be different from the example output above.

Fix

Configure Ubuntu 22.04 LTS to audit activities performed during nonlocal maintenance and diagnostic sessions.

Add or modify the following line in the "/etc/audit/rules.d/stig.rules" file:

-w /var/log/sudo.log -p wa -k maintenance

To reload the rules file, issue the following command:

```
$ sudo augenrules --load
```

Note: The "-k " at the end of the line gives the rule a unique meaning to help during an audit investigation. The does not need to match the example above.

V-260650

Title

Ubuntu 22.04 LTS must implement NIST FIPS-validated cryptography to protect classified information and for the following: To provision digital signatures, to generate cryptographic hashes, and to protect unclassified information requiring confidentiality and cryptographic protection in accordance with applicable federal laws, Executive Orders, directives, policies, regulations, and standards.

Description

Use of weak or untested encryption algorithms undermines the purposes of utilizing encryption to protect data. The operating system must implement cryptographic modules adhering to the higher standards approved by the federal government since this provides assurance they have been tested and validated.

Satisfies: SRG-OS-000396-GPOS-00176, SRG-OS-000478-GPOS-00223

Rule Check

Verify the system is configured to run in FIPS mode by using the following command:

```
$ grep -i 1 /proc/sys/crypto/fips_enabled
1
```

If a value of "1" is not returned, this is a finding.

Fix

Configure Ubuntu 22.04 LTS to run in FIPS mode. Add "fips=1" to the kernel parameter during Ubuntu 22.04 LTS install.

Enabling a FIPS mode on a pre-existing system involves a number of modifications to Ubuntu 22.04 LTS. Refer to the Ubuntu Pro security certification documentation for instructions.

A subscription to the "Ubuntu Pro" plan is required to obtain the FIPS Kernel cryptographic modules and enable FIPS.

Note: Ubuntu Pro security certification instructions can be found at: https://ubuntu.com/security/certifications/docs/fips-enablement

---