EAL2 Evaluated Configuration Guide for Red Hat Virtualization 4.3

November 8, 2021; v0.9

atsec is a trademark of atsec information security GmbH

Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Java and all Java-based products are trademarks of Sun Microsystems, Inc., in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds.

UNIX is a registered trademark of The Open Group in the United States and other countries.

All other trademarks are property of their respective owners.

This document is provided AS IS with no express or implied warranties. Use the information in this document at your own risk.

Contents

Chapter 1

Introduction

1.1 Purpose of this document

The Red Hat Virtualization (RHV) distribution is designed to provide a secure and reliable operating system for a variety of purposes. Because security requirements obviously depend on the applications and environment, it is not possible to simply certify that the system is "secure", a more precise definition is needed.

The Common Criteria (CC) provides a widely recognized methodology for security certifications. A CC evaluation is fundamentally a two-step process, consisting of defining the "security target" which describes the features that are to be evaluated, and then testing and verifying that the system actually implements these features with a sufficient level of assurance.

This document is a security guide that explains how to set up the evaluated configuration, and provides information to administrators and ordinary users to ensure secure operation of the system. It is intended to be self-contained in addressing the most important issues at a high level, and refers to other existing documentation where more details are needed.

The document primarily addresses administrators, but the section "Security guidelines for users" is intended for ordinary users of the system as well as administrators.

Knowledge of the Common Criteria is not required for readers of this document.

1.2 How to use this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 (http://www.ietf.org/rfc/rfc2119.txt)

Note that this document avoids the terms "SHOULD" and "SHOULD NOT" that are defined in RFC 2119. Requirements are either absolute (and marked with MUST and equivalent terms), or entirely optional (in the sense of not affecting required security functions) and marked with RECOMMENDED, MAY or OPTIONAL.

If you follow the requirements in this document when setting up and using the system, your configuration will match the evaluated configuration. Certain configuration options are marked as OPTIONAL and you MAY modify them as needed, but you MUST NOT make other changes, because they will make the system fail to match the evaluated configuration.

Of course, you MUST always use common sense. This document is not a formal specification, and legitimate reasons can exist to modify the system setup in ways not described here if that is necessary for the system to fulfill its intended purpose. Specifically, applying security patches released by the vendor is strongly RECOMMENDED even though that will cause a deviation from the evaluated configuration.

In cases where the requirements and recommendations in this document conflict with those in other sources (such as the online documentation), the information in this Configuration Guide has higher precedence. You MUST follow the steps described here to reach the evaluated configuration, even if other documentation describes different methods.

The usual convention is used in this guide when referring to manual pages that are included in the software distribution. For example, the notation *ls*(1) means that running the `man -S 1 ls` command will display the manual page for the *ls* command from section one of the installed documentation. In most cases, the `-S` flag and the section number can be omitted from the command, they are only needed if pages with the same name exist in different sections,

1.3 Requirements and assumptions

1.3.1 What is a CC compliant system?

A system can be considered to be "CC compliant" if it matches an evaluated and certified configuration. This implies various requirements concerning hardware and software, as well as requirements concerning the operating environment, users, and the ongoing operating procedures.

Strictly speaking, an evaluation according to the CC represents the results of investigation of the security properties of the target system according to defined guidelines. It should not be considered as a guarantee for fitness for any specific purpose, but should provide help in deciding the suitability of the system considering how well the intended use fits the described capabilities. It is intended to provide a level of assurance about the security functions that have been examined by a neutral third party.

The software MUST match the evaluated configuration. In the case of an operating system, this also requires that the installed kernel, system, and application software are the same.

Stated requirements concerning the operating environment MUST be met. Typical requirements include a secure location for the hardware (protected from physical access by unauthorized persons), as well as restrictions concerning permitted network connections.

The operation of the system MUST be in agreement with defined organizational security policies, to ensure that actions by administrators and users do not undermine the system's security.

1.3.2 Hardware requirements

The hardware MUST be the following hardware system. This entire document applies to this hardware system unless explicitly noted.

Systems based x86 64bit Intel Xeon processors

Running the certified software on other similar hardware might result in an equivalent security level, but the certification does not apply if the hardware is different from that used for the testing processes during the evaluation.

Note, the proper operation of all aspects of the software is only ensured when using the aforementioned hardware systems as several hardware mechanisms which may not be present in other systems are vital for the security of the system.

Please refer to section §2.1 "Supported hardware" for more information about additional hardware supported for use with the evaluated configuration.

1.3.3 Requirements for the system's environment

The security target covers one or more systems running RHV, networked in a non-hostile network, with a well-managed and non-hostile user community. It is not intended to address the needs of an Internet-connected server, or the case where services are to be provided to potentially hostile users.

It is assumed that no undocumented security critical side affects and no compromisation prior to the installation are present. Further, if covert storage or timing channels exist, sufficient security is assumed to be used relative to accessible IT assets to outweigh the risk of a security policy violation. It is also assumed that physical controls in place would alert the system authorities to the physical presence of attackers within the controlled space.

You MUST set up the server (or servers) in a physically secure environment, where they are protected from theft and manipulation by unauthorized persons.

You MUST ensure that all connections to peripheral devices and all network connections are protected against tampering, tapping and other modifications. Using the secured protocol of SSHv2, TLS or IPSEC is considered sufficient protection for network connections. All other connections must remain completely within the physically secure server environment.

1.3.4 Requirements for connectivity

All components in the network such as routers, switches, and hubs that are used for communication are assumed to pass the user data reliably and without modification. Translations on protocols elements (such as NAT) are allowed as long as those modifications do not lead to a situation where information is routed to somebody other than the intended recipient system. Network and peripheral cabling must be approved for the transmittal of the most sensitive data held by the system.

Any other systems with which the system communicates MUST be under the same management control and operate under the same security policy constraints.

Be aware that information passed to another system leaves the control of the sending system, and the protection of this information against unauthorized access needs to be enforced by the receiving system. If an organization wants to implement a consistent security policy covering multiple systems on a network, organizational procedures MUST ensure that all those systems can be trusted and are configured with compatible security configurations enforcing an organization wide security policy. How to do this is beyond the scope of this Configuration Guide. If you set up a communication link to a system outside your control, please keep in mind that you will not be able to enforce any security policy for any information you pass to such a system over the communication link or in other ways (for example, by using removable storage media).

1.3.5 Requirements for administrators

There MUST be one or more competent individuals who are assigned to manage the system and the security of the information it contains.

The system administrative personnel MUST NOT be careless, willfully negligent, or hostile, and MUST follow and abide by the instructions provided by the administrator documentation.

Every person that has the ability to perform administrative actions by switching to root has full control over the system and could, either by accident or deliberately, undermine security features of the system and bring it into an insecure state. This Configuration Guide provides the basic guidance how to set up and operate the system securely, but is not intended to be the sole information required for a system administrator to learn how to operate Linux securely.

It is assumed, within this Configuration Guide, that administrators who use this guide have a good knowledge and understanding of operating security principles in general and of Linux administrative commands and configuration options in particular. We strongly advise that an organization that wants to operate the system in the evaluated configuration nevertheless have their administrators trained in operating system security principles and RHV security functions, properties, and configuration.

Every organization needs to trust their system administrators not to deliberately undermine the security of the system. Although the evaluated configuration includes audit functions that can be used to make users accountable for their actions, an administrator is able to stop the audit subsystem and reconfigure it such that his actions no longer get audited. Well trained and trustworthy administrators are a key element for the secure operation of the system. This Configuration Guide provides the additional information a system administrator should obey when installing, configuring and operating the system in compliance with the requirements defined in the Security Target for the Common Criteria evaluation.

The above stated assumptions imply that the DAC permissions of system directories, system binary files and their configuration files are left unchanged.

To ensure the integrity of the system, you MUST schedule periodical reviews of the system operation and system integrity. For example, an integrity verification using the `rpm` tool may be invoked. Another possibility of validating the integrity of the system is the use of `aide`.

1.3.6 Requirements for the system's users

The security target addresses the security needs of cooperating users in a benign environment, who will use the system responsibly to fulfill their tasks.

Authorized users possess the necessary authorization to access at least some of the information managed by the system and are expected to act in a cooperating manner in a benign environment.

Note that system availability is *not* addressed in this evaluation, and a malicious user could disable a server through resource exhaustion or similar methods.

The requirements for users specifically include:

- User accounts MUST be assigned only to those users with a need to access the virtual machines protected by the system, and who MUST be sufficiently trustworthy not to abuse those privileges. For example, the system cannot prevent data from being intentionally redistributed to unauthorized third parties by an authorized user.
- Users are trusted to accomplish some task or group of tasks within a secure IT environment by exercising complete control over their virtual machines.
- All users of the system MUST be sufficiently skilled to understand the security implications of their actions, and MUST understand and follow the requirements listed in section §6 "Security guidelines for users" of this guide. Appropriate training MUST be available to ensure this.

It is part of your responsibility as a system administrator to verify that these requirements are met, and to be available to users if they need your help in maintaining the security of their data.

Chapter 2

Installation

The evaluation covers a fresh installation of RHV Version 4.3, on one of the supported hardware platforms as defined in section §1.3.2 "Hardware requirements" of this guide.

The evaluated configuration MUST be the only operating system installed on the server.

2.1 Supported hardware

You MAY attach the following peripherals without invalidating the evaluation results. Other hardware MUST NOT be installed in or attached to the system.

- Any storage devices and backup devices supported by the operating system.
- All Ethernet network adapters supported by the operating system. Other types of WAN adapters are not part of the evaluated environment.
- Operator console consisting of a keyboard, video monitor, and optionally mouse. Additionally, you MAY directly attach supported serial terminals, but *not* other types of remote access terminals which guarantees that the console is not remotely accessible (the serial terminal with a directly connected console is not considered a remote access).

2.2 Installation Considerations

The evaluated configuration installs an administrative interface with the Red Hat Virtualization Manager. Although the manager offers a role-based system to restrict the actions of administrators, this evaluation considers all users that have access to the Red Hat Virtualization Manager as administrative users.

The role-based access control was not subject to evaluation.

Therefore, the installation and the operation of the system must guarantee that only administrators are able to interact with the Red Hat Virtualization Manager. This can be achieved with one or more approaches from the following list:

- Only administrative users have an account on the Red Hat Virtualization Manager. Please note, the installation allows to link the RHV with a central administration system like an LDAP server or an Microsoft Active Directory. When using such central credential repositories, all users that are configured to allow access to the Red Hat Virtualization Manager are considered to be administrators to the system.

- The Red Hat Virtualization Manager may only be accessible via an administrative LAN. This implies that the network interface through which the manager is accessible is connected to a network which is dedicated to administrative user. This may be achieved, for example, by using dedicated physical cabling or by using VLANs.

Non-administrative users are only users that can interact with virtual machines they are granted access to. This includes access to the console made accessible via VNC or serial console of the virtual machine.

2.3 Selection of install options and packages

This section describes the detailed steps to be performed when installing the RHV operating system on the target server.

All settings listed here are REQUIRED unless specifically declared otherwise.

2.3.1 Prerequisites for installation

You will need the following components to install a system in the evaluated configuration as explained in the following sections:

- The target system that will be installed, refer to section §1.3.2 "Hardware requirements" of this guide for the list of supported hardware.
- All hardware requirements outlined in chapter 2 "Requirements" of the "Installing Red Hat Virtualization as a self-hosted engine using the command line" document at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/ MUST be used.

2.3.2 Preparing for installation

You MUST download the distribution ISO images from the Red Hat Network on a separate Internet-connected computer, and either burn CD-Rs from them, or make the contents available on a file server via NFS, HTTP, or FTP. The download location https://access.redhat.com/downloads/content/415/ contains links to the platform-specific image.

Note, you need to login to the Red Hat Customer Portal to access the link.

That link takes the user to a page where the product variant and version must be selected. As product variant **Red Hat Virtualization Manager** and as version **4.3** MUST be selected.

Furthermore the following ISO image MUST be selected for installation.

- Hypervisor Image 4.3.17 EUS

The SHA-256 checksum for the image is *f6267ccee75c8cfb027f891e20cd38d1ac3da75d43740e11f3ec7b576f06a6e2*.

After downloading the ISO image, you MUST verify that the SHA256 checksum of the image file is correct. The checksums are shown on the RHN web page, please verify that the web page is encrypted (https:// URL) and has a valid certificate. Then run sha256sum *.iso to view the checksum of the downloaded image, and compare it with the one shown on the web page.

You MUST perform all preparation steps for the system described in chapter 3 "Preparing Storage for Red Hat Virtualization" of the "Installing Red Hat Virtualization as a self-hosted engine using the command line" document at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/ as necessary for the deployment environment.

2.3.3 FIPS 140-2

RHV provides a central flag to switch various cryptographic libraries as well as cryptographic applications into FIPS 140-2 compliant mode. Although the FIPS 140-2 mode is out of scope for a Common Criteria evaluation, both modes work well together.

If you are not concerned with the FIPS 140-2 compliant operation of cryptographic mechanisms, you MAY skip this section.

The Security Policy documents of the various FIPS 140-2 modules delivered with RHV can be obtained at the NIST web page at http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm. These documents provide the guidance for bringing the respective module into the FIPS 140-2 compliant mode. All Security Policy modules document how to achieve that mode by configuring RHV at runtime.

However, the following aspects for the FIPS 140-2 modules cannot be handled at runtime. If complete FIPS 140-2 compliance is requested, the following considerations should be made:

- The host keys used by the OpenSSH server are generated during the first boot of the installed system. If these keys shall be generated in complete compliance with the FIPS 140-2 requirements, the installation system must be booted with *fips=1*.

Using that *fips=1* boot flag ensures the following:

- The kernel is booted with the *fips=1* kernel command line flag as required by the Security Policy documents for the different FIPS 140-2 modules. When using these modules during the initial installation time, all operations are performed compliant to the FIPS 140-2 requirements.
- The kernel command line applied to the newly installed system also contains the *fips=1* as well as the *boot=* flag as outlined in the Security Policy documents for the different FIPS 140-2 modules to ensure the newly installed system is booted in FIPS 140-2 mode.
- The `dracut` boot mechanism contains the `dracut-fips` extension which ensures that the boot-time self-tests as well as integrity tests for the kernel are performed.

Therefore, after booting the installation system with the *fips=1* flag, all system setup configuration steps outlined in the various Security Policy documents are already covered.

Note that boot procedures with *fips=1* may take longer and sometimes much longer than regular boot procedures as the FIPS 140-2 self tests, integrity tests and random number generator seeding procedures must be performed.

For more information about the FIPS 140-2 compliant operation of RHV, please consult the Security Policy document applicable for the intended cryptographic module.

2.4 Installation

It is RECOMMENDED that you disconnect all network connections until the post-install system configuration is finished. You MAY use a network if required for the installation (for example when using a NFS or HTTP network server instead of CD-ROMs). If you do use a network, you MUST ensure that this network is secure.

RHV MUST be installed as self-hosted engine which can be deployed either from a Red Hat Virtualization Host or a Red Hat Enterprise Linux host. This is typically done by inserting first CD and boot from CD-ROM. When obtaining the one of the following installation guides via the Internet, you should use an SSL-protected HTTP connection to ensure the integrity and authenticity of the documentation.

To install the Red Hat Virtualization Host you MUST follow the steps in section 4.1. "Installing Red Hat Virtualization Hosts" of the "Installing Red Hat Virtualization as a self-hosted engine using the command line" documentation file at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/. To register the system and receive updates, the Red Hat Virtualization Host MUST be registered with the Content Delivery Network or with Red Hat Satellite 6 (Section 4.1.1. of the previous mentioned document).

To install the Red Hat Enterprise Linux host the description in "Installing Red Hat Virtualization as a self-hosted engine using the command line" section 4.2. "Installing Red Hat Enterprise Linux hosts" of the documentation at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/ MUST be followed.

2.4.1 Verify Installation

To verify the TOE was successfully installed you MUST use one of the following methods.

Either run the following command and verify that "rhvh-4.3.17" is used

```
nodectl info
```

or verify that "Red Hat Virtualization Host" in version "4.3.17" is used by executing

```
cat /etc/os-release
```

2.4.2 Post-Installation Steps

After the installation is complete, the following step MUST be performed to harden the system and to be consistent with the evaluated configuration.

Log into the console of RHV as root and remove the tcpdump package:

```
yum remove tcpdump
```

Furthermore, the SCTP module MUST be disabled by preventing the module from loading with the following instruction:

```
echo "install sctp /bin/true" >> /etc/modprobe.d/disable-sctp.conf
```

Afterwards a reboot of the system is necessary to apply all changes.

Chapter 3

Secure initial system configuration

After the initial installation using the procedure described in the previous chapter, the operating system is already in the evaluated configuration.

The system does not define audit rules as there is no universally applicable default for this. Please refer to section §5.3 "Configuring the audit subsystem" of this guide for more information.

Chapter 4

System operation

To ensure that the systems remains in a secure state, special care MUST be taken during system operation.

4.1 Gaining administrative access

The Red Hat Virtualization Manager is installed as part of the system as the management interface to control resources and virtual machines. This manager is the administrative interface that is to be used by administrators.

Even though, the interface offered by the Red Hat Virtualization Manager offers various roles to restrict the rights of users interacting with this manager, this role-mechanism was not subject to evaluation. Thus, all users interacting with the manager must be considered as trusted administrators irrespective of their assigned role.

You MUST ensure that only administrators are allowed to access this administrative interface as outlined in section §2.2. This is achieved using the user specification as outlined in *Administration Guide* section 1.2 in conjunction with *Virtual Machine Management Guide* section 2.1.

4.2 Virtualization Configuration

The Red Hat Virtualization Manager allows the management of resources that can be used to create virtual machines with. The entire management of these resources is outside of the CC evaluation. All management operations are allowed to be conducted, though. Therefore, the administration guidance accompanied with the Red Hat Virtualization solution is applicable.

The RHV guidance covering the configuration of Linux containers is provided the guide *Virtual Machine Management Guide* as well as *Administration Guide*.

To operate the virtualization solution compliant to restrictions stipulated by the evaluation, the following sections outline configuration requirements to comply with these restrictions.

Any settings allowed and documented in the *Virtual Machine Management Guide*, *Administration Guide*, and *Data Warehouse Guide* that are not subject to restrictions mentioned in the following are allowed.

Please note that the Cockpit is not allowed, only the Red Hat virtualization manager is permitted in the evaluated configuration.

4.3 Storage Configuration

Albeit RHV is capable of supporting many different storage solutions to provide disk space to virtual machines, virtual machines are only allowed with virtual disk storage. The reason is that only for virtual disks RHV ensures that all previously existing data is destroyed during reallocation of disk space.

It is permissible to use any type of storage systems including storage networks as long as virtual disks are supported and used for virtual machines.

Other types of disk storage are not allowed to be assigned to virtual machines.

4.4 Device Passthrough Configuration

RHV supports a configuration of virtual machines where one virtual machine has exclusive access to a particular physical hardware. Such hardware includes PCI devices and USB devices. Such configuration is also referred to as "passthrough" configuration of physical devices.

A virtual machine that accesses a passthrough device must implement its own device driver to access the device.

PCI passthrough MUST NOT be configured as it is a security issue. A virtual machine that has full access to a physical PCI device has potentially the capability to replace the PCI device BIOS if it is updatable. As the x86 boot sequence performs a PCI device enumeration which invokes the PCI device BIOS if present. This implies in case of PCI passthrough that a virtual machine controls code that is executed with hardware privilege during boot time of the host which is considered a security violation.

USB device passthrough is permitted as such USB devices are not affected by the issue explained for PCI devices. Yet, the administrator MUST perform a sanitization operation when a USB device is to be reassigned from one virtual machine to another. If the administrator is unsure how to perform a complete sanitization operation, the respective USB device MUST NOT be re-assigned.

4.5 Locking and unlocking of user accounts

The *pam_faillock.so* module maintains a list of failed authentication attempts per user during a specified interval and locks the account in case there were more than deny consecutive failed authentications. It stores the failure records into per-user files in the tally directory.

The *faillock* command is an application which can be used to examine and modify the contents of the the tally files. It can display the recent failed authentication attempts of the username or clear the tally files of all or individual usernames.

Please note that the order is very important while adding these lines to the pam configuration files */etc/pam.d/password-auth* and */etc/pam.d/system-auth*. As the importance of the order is not described in depth in `faillock(8)` an example with the three added lines

```
auth            required          pam_faillock.so preauth silent audit
even_deny_root deny=3 unlock_time=600
auth            [default=die] pam_faillock.so authfail audit
even_deny_root deny=3 unlock_time=600
account         required          pam_faillock.so
```

to */etc/pam.d/password-auth* is shown below. This also applies in the same way to the file */etc/pam.d/system-auth*.

```
auth        required      pam_env.so
auth        required      pam_faillock.so preauth silent audit
even_deny_root deny=3 unlock_time=600
auth        required      pam_faildelay.so delay=2000000
auth        sufficient    pam_unix.so nullok try_first_pass
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        [default=die] pam_faillock.so authfail audit
even_deny_root deny=3 unlock_time=600
auth        required      pam_deny.so

account     required      pam_faillock.so
account     required      pam_unix.so
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     required      pam_permit.so

password    requisite     pam_pwquality.so try_first_pass
local_users_only retry=3 authtok_type=
password    sufficient    pam_unix.so sha512 shadow nullok
try_first_pass use_authtok

password    required      pam_deny.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
-session    optional      pam_systemd.so
session     [success=1 default=ignore] pam_succeed_if.so service
in crond quiet use_uid
session     required      pam_unix.so
```

See for more information, especially for the possible options, the manpage of the `faillock(8)` module.

4.6 Setting of password requirements

The pam_pwquality module can be plugged into the password stack of a given service to provide some plug-in strength-checking for passwords.

The action of this module is to prompt the user for a password and check its strength against a system dictionary and a set of rules for identifying poor choices.

The first action is to prompt for a single password, check its strength and then, if it is considered strong, prompt for the password a second time (to verify that it was typed correctly on the first occasion). All being well, the password is passed on to subsequent modules to be installed as the new authentication token.

See for more information the manpage of the `pam_pwquality(8)` module.

Chapter 5

Monitoring, Logging & Audit

5.1 Reviewing the system configuration

It is RECOMMENDED that you review the system's configuration at regular intervals to verify if it still agrees with the evaluated configuration. This primarily concerns those processes that run with root privileges.

The permissions of the device files */dev/** MUST NOT be modified.

In particular, review settings in the following files and directories to ensure that the contents and permissions have not been modified:

```
/etc/audit/*
/etc/cron.{ weekly hourly daily monthly}
/etc/cron.allow
/etc/cron.d/*
/etc/cron.deny
/etc/crontab
/etc/group
/etc/gshadow
/etc/hosts
/etc/ipsec.conf
/etc/ipsec.d/*
/etc/ipsec.secrets
/etc/ld.so.conf
/etc/localtime
/etc/login.defs
/etc/modprobe.conf
/etc/netlabel.rules
/etc/pam.d/*
/etc/passwd
/etc/systemd/*
/etc/securetty
/etc/security/opasswd
/etc/selinux/config
/etc/selinux/mls/contexts/
/etc/selinux/mls/modules/
/etc/selinux/mls/policy/
/etc/selinux/mls/setrans.conf
/etc/selinux/mls/seusers
```

```
        /etc/selinux/semanage.conf
        /etc/shadow
        /etc/ssh/*
        /etc/sysconfig/*
        /etc/sysctl.conf
        /etc/sssd/*
        /var/log/lastlog
        /var/run/faillock/*
        /var/spool/cron/root
```

Use the commands `lastlog` as well as `last` to detect unusual patterns of logins.

Also verify the output of the following commands (run as root):

```
        crontab -l
        find / \( -perm -4000 -o -perm -2000 \) -ls
        find / \( -type f -o -type d -o -type b \) -perm -0002 -ls


        find /bin /boot /etc /lib /sbin /usr \
                ! -type l \( ! -uid 0 -o -perm +022 \)
```

5.2 System logging and accounting

System log messages are stored in the */var/log/* directory tree. Most of the logs are stored in plain text format when logged through the *rsyslogd*(8) program, which MAY be configured via the */etc/rsyslog.conf* file.

In addition, *systemd* and its helper application store data in binary format in the */var/log/* directory tree. Those logs can be reviewed using the `journalctl` application. See *journalctl*(8) for more details.

The *logrotate*(8) utility, launched from */etc/cron.daily/logrotate*, starts a fresh log file every week or when they reach a maximum size and automatically removes or archives old log files. You MAY change the configuration files */etc/logrotate.conf* and */etc/logrotate.d/** as required.

In addition to the *rsyslog* messages, various other log files and status files are generated in */var/log* by other programs:

File	Source
audit/	Default audit log storage
boot.log	Messages from system startup
journal/	Messages from systemd
maillog	Written by syslog, contains messages from the MTA (postfix)
messages	Written by syslog, contains messages from su and ssh
secure	Security related messages (for example from PAM)

Please see *rsyslog.conf*(5) and *rsyslogd*(8) man pages for details on syslog configuration.

Please see *journald.conf*(5) for details on configuring the *systemd* logging facility.

The *ps*(1) command can be used to monitor the currently running processes. Using `ps faux` will show all currently running processes and threads.

5.3 Configuring the audit subsystem

The audit subsystem implements a central monitoring solution to keep track of security relevant events, such as changes and change attempts to security critical files.

This is accomplished through two separate mechanisms. All system calls are intercepted, and the kernel writes the parameters and return value to the audit log for those calls that are marked as security relevant in the configuration of audit rules (filter). In addition, some trusted programs contain audit-specific code to write audit trails of the actions they are requested to perform.

Please refer to the *auditd*(8), *auditd.conf*(5), audit.rules(7) and *auditctl*(8) man pages for more information.

5.3.1 Selecting the events to be audited

You MAY make changes to the set of system calls and events that are to be audited. The evaluation requires that the system has the *capability* to audit security relevant events, but it is up to you to choose how you want to use these capabilities. It is acceptable to turn off system call auditing completely even in an evaluated configuration, for example on a pure application server with no interactive users on the system.

The man page *audit.rules*(7) provides a number of tips as well as auditing strategies. In addition *augenrules*(8) provides additional information on the audit rule generation.

You MAY selectively disable and enable auditing for specific events or users as required by modifying the */etc/audit/rules.d/audit.rules* file.

It is RECOMMENDED that you monitor use of the *semodule*(8) tool to keep track of administrative changes to optional security policy modules:

```
-w /usr/sbin/semodule
```

It is RECOMMENDED that you always reconfigure the audit system by modifying the */etc/audit/rules.d/audit.rules* file and then running the following command to reload the audit rules:

```
systemctl restart auditd
```

This procedure ensures that the state of the audit system always matches the content of the */etc/audit/rules.d/audit.rules* file. You SHOULD NOT manually add and remove audit rules and watches on the command line as those changes are not persistent.

Note that reloading audit rules involves initially deleting all audit rules, and for a short time the system will be operating with no or only a partial set of audit rules. It is RECOMMENDED to make changes to the audit rules when no users are logged in on the system, for example by rebooting the system to activate the changes. The single user mode should not be used for this operation as it does not deconfigure network interfaces and therefore does not prevent users from logging in.

Please refer to the *auditctl*(8) man page for more details.

5.3.2 Reading and searching the audit records

Use the *ausearch*(8) tool to retrieve information from the audit logs. The information available for retrieval depends on the active filter configuration. If you modify the filter configuration, it is RECOMMENDED keeping a datestamped copy of the applicable configuration with the log files for future reference.

For example:

```
# search for events with a specific login UID
ausearch -ul jdoe

# search for events by process ID
ausearch -p 4690
```

Please refer to the *ausearch*(8) man page for more details. In addition, the *audit.rules*(7) man page contains additional hints on implementing effective audit trail searches.

For some system calls on some platforms, the system call arguments in the audit record can be slightly different than you may expect from the program source code due to modifications to the arguments in the C library or in kernel wrapper functions. For example, the *mq_open*(3) glibc library function strips the leading '/' character from the path argument before passing it to the *mq_open*(2) system call, leading to a one character difference in the audit record data. Similarly, some system calls such as *semctl*(2), *getxattr*(2), and *mknodat*(2) can have additional internal flags automatically added to the flag argument. These minor modifications do not change the security relevant information in the audit record.

Of course, you can use other tools such as plain *grep*(1) or scripting languages such as *awk*(1), *python*(1) or *perl*(1) to further analyze the text audit log file or output generated by the *ausearch* tool.

5.3.3 Starting and stopping the audit subsystem

If the audit daemon is terminated, no audit events are saved until it is restarted. To avoid lost audit records when you have modified the filter configuration, you MUST use the command `service auditd reload` to re-load the filters.

You MUST NOT use the *KILL* signal (-9) to stop the audit daemon, doing so would prevent it from cleanly shutting down.

It is RECOMMENDED that you add the kernel parameter `audit=1` to your boot loader configuration file to ensure that all processes, including those launched before the *auditd* service, are properly attached to the audit subsystem. Please refer to the documentation of your boot loader for more details.

5.3.4 Storage of audit records

The default audit configuration stores audit records in the */var/log/audit/audit.log* file. This is configured in the */etc/audit/auditd.conf* file. You MAY change the *auditd.conf* file to suit your local requirements.

It is RECOMMENDED that you configure the audit daemon settings appropriately for your local requirements, for example by changing the log file retention policy to never delete old audit logs with the following setting in the */etc/audit/auditd.conf* file:

```
max_log_file_action = KEEP_LOGS
```

The most important settings concern handling situations where the audit system is at risk of losing audit information, such as due to lack of disk space or other error conditions. You MAY choose actions appropriate for your environment, such as switching to single user mode (action `single`) or shutting down the system (action `halt`) to prevent auditable actions when the audit records cannot be stored.

Warning: Switching to single user mode does not automatically kill all user processes when using the system default procedure. You MAY kill processes of users by using `pkill -u`. Please note that system services SHOULD NOT be terminated. Depending on your local policy, you MAY need to shut down KVM guests. As these KVM guests act like normal processes on the Linux host system, the same commands to terminate these processes may be used.

Halting the system is RECOMMENDED and most certain way to ensure all user processes are stopped. The following settings are RECOMMENDED in the */etc/audit/auditd.conf* file if a fail-secure audit system is required:

```
admin_space_left_action = SINGLE
disk_full_action = HALT
disk_error_action = HALT
```

It is RECOMMENDED that you configure appropriate disk space thresholds and notification methods to receive an advance warning when the space for audit records is running low.

It is RECOMMENDED that you use a dedicated partition for the */var/log/audit/* directory to ensure that *auditd* has full control over the disk space usage with no other processes interfering.

Please refer to the *auditd.conf*(5) man page for more information about the storage and handling of audit records.

5.3.5 Reliability of audit data

You MAY choose an appropriate balance between availability of the system and secure failure mode in case of audit system malfunctions based on your local requirements.

You MAY configure the system to cease all processing immediately in case of critical errors in the audit system. When such an error is detected, the system will then immediately enter "panic" mode and will need to be manually rebooted. To use this mode, add the following line to the */etc/audit/rules.d/audit.rules* file:

```
-f 2
```

Please refer to the *auditctl*(8) man page for more information about the failure handling modes.

You MAY edit the */etc/libaudit.conf* file to configure the desired action for applications that cannot communicate with the audit system. Please refer to the *get_auditfail_action*(3) man page for more information.

auditd writes audit records using the normal Linux filesystem buffering, which means that information can be lost in a crash because it has not been written to the physical disk yet. Configuration options control how *auditd* handles disk writes and allow the administrator to choose an appropriate balance between performance and reliability.

Any applications that read the records while the system is running will always get the most current data out of the buffer cache, even if it has not yet been committed to disk, so the buffering settings do not affect normal operation.

The default setting is `flush = DATA`, ensuring that record data is written to disk, but metadata such as the last file time might be inconsistent.

The highest performance mode is `flush = none`, but be aware that this can cause loss of audit records in the event of a system crash.

If you want to ensure that auditd always forces a disk write for each record, you MAY set the `flush = SYNC` option in */etc/audit/auditd.conf*, but be aware that this will result in significantly reduced performance and high strain on the disk.

A compromise between crash reliability and performance is to ensure a disk sync after writing a specific number of records to provide an upper limit for the number of records lost in a crash. For this, use a combination of `flush = INCREMENTAL` and a numeric setting for the `freq` parameter, for example:

```
flush = INCREMENTAL
freq = 100
```

The audit record files are *not* protected against a malicious administrator, and are not intended for an environment where the administrators are not trustworthy.

Chapter 6

Security guidelines for users

Users are only allowed to interact with the virtual machine assigned to them as well as the console and serial console of this virtual machine. Access to the consoles is given by administrators.

The console access is made available as outlined in *Virtual Machine Management Guide* sections 2.2.2 and 2.2.3.

Chapter 7

Appendix

7.1 Online Documentation

If there are conflicting recommendations in this guide and in one of the sources listed here, the Configuration Guide has precedence concerning the evaluated configuration.

RHV 4.3 Guidance: https://access.redhat.com/documentation/en-US/Red_Hat_Virtualization/4.3/