SUSE

Understanding SELinux Basics

WHAT?

This article provides basic information about Security-Enhanced Linux.

WHY?

You want to understand SELinux and how to configure it on SUSE Linux Micro.

EFFORT

It takes approximately 40 minutes to read this article.

Publication Date: 07 Nov 2025

Contents

1 About SELinux

SELinux was developed as an additional Linux security solution that uses the security framework in the Linux kernel. The purpose was to allow for a more granular security policy that goes beyond the standard Discretionary Access Controls (DAC), the traditional file permissions of owner/group/world, and read/write/execute.

SELinux implements Mandatory Access Control (MAC). Each process and system resource has a security label called an SELinux context. Also called an SELinux label, it is an identifier that contains information about a system object, such as files and network sockets. These labels are used to make access control decisions.

The default action of SELinux is to deny any access. SELinux allows only actions that are specifically allowed in the SELinux policy. Another feature of SELinux that increases security is that SELinux allows strict confinement of processes up to the point where the processes cannot access files of other processes on the same system.

SELinux was designed to enhance existing security solutions, not to replace them. For example, discretionary access control (DAC) is still applied, even if the system is using SELinux. If DAC denies access first, SELinux is then not used as the access was already blocked by another mechanism. SELinux is present on the pre-built images by default.

2 Why use SELinux?

The benefits of using SELinux are:

Labeling

SELinux uses security labels, or contexts, to identify and classify processes, files, and other system objects. This labeling system enables precise enforcement of access control policies.

Detailed access control

Administrators can define very specific rules for how processes interact with files, directories, and other system resources. This enables granular control over system security.

Mitigate zero-day exploits

A zero-day exploit is a cyber attack that targets an unknown software vulnerability. Attackers gain access to a system before a patch or fix is available. SELinux can help mitigate the impact of zero-day exploits by restricting the action a compromised application can take.

Detailed auditing

SELinux provides detailed audit logs, which can be used to track and analyze security events to help identify and mitigate potential threats.

Enhanced system integrity

SELinux helps to maintain the integrity of the operating system and the data by enforcing consistent access controls.

3 SELinux modes

SELinux can run in `enforcing` or `permissive` mode:

`enforcing`

The security policy is enforced. Any access that is not explicitly allowed by the policy is denied.

`permissive`

SELinux is active, the security policy is loaded, the file system is labeled and access denial entries are logged. However, the policy is not enforced, and no access is denied.

For information about switching between SELinux modes, refer to *Section 3.1, “Changing the SELinux mode”*.

3.1 Changing the SELinux mode

You can switch the SELinux mode temporarily or permanently.

3.1.1 Verifying the active SELinux mode

To verify the mode, run the following command:

```
# getenforce
```

The command should return `enforcing` or `permissive`.

3.1.2 Changing the SELinux mode temporarily

To set SELinux to the `permissive` or `enforcing` mode temporarily, use the command **`setenforce`**.

The **`setenforce`** command has the following syntax:

```
# setenforce MODE_ID
```

where *`MODE_ID`* is *`0`* for the `permissive` mode or *`1`* for the `enforcing` mode.

Remember that you cannot disable SELinux using the **`setenforce`** command.

3.1.3 Changing the SELinux mode permanently

To perform changes to the SELinux mode that persists after a reboot, edit the `/etc/selinux/config` configuration file. If SELinux is causing issues on your system, switch to the `permissive` mode and debug your system.

In the file `/etc/selinux/config`, change the value of `SELINUX` to `permissive`, or `enforcing` as follows:

```
SELINUX=permissive
```

The changes in the file are applied after the next reboot.

3.1.4 Switching from permissive to enforcing mode

When switching from permissive to enforcing mode, it is essential that the security context is reset on all files. They might have changed during the time in permissive mode.

Some additional steps are necessary to ensure that your system works properly in `enforcing` mode:

1. Reset the security context on the running system:

   ```
   # restorecon -R /
   ```

   Or mark a reset that is done during boot:

   ```
   # touch /etc/selinux/.autorelabel
   ```

2. Switch to enforcing mode by setting `SELINUX=enforcing` in the `/etc/selinux/config`.

3. Reboot the system and log in again.

4. Run the **`sestatus -v`** command. It should give you an output similar to the following one:

   ```
   # sestatus -v
   SELinux status:                 enabled
   ```

```
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33

Process contexts:
Current context:                unconfined_u:unconfined_r:unconfined_t:s0-
s0:c0.c1023
Init context:                   system_u:system_r:init_t:s0
/sbin/agetty                    system_u:system_r:getty_t:s0-s0:c0.c1023
/usr/sbin/sshd                  system_u:system_r:sshd_t:s0-s0:c0.c1023

File contexts:
Controlling terminal:           unconfined_u:object_r:user_devpts_t:s0
/etc/passwd                     system_u:object_r:passwd_file_t:s0
/etc/shadow                     system_u:object_r:shadow_t:s0
/bin/bash                       system_u:object_r:shell_exec_t:s0
/bin/login                      system_u:object_r:login_exec_t:s0
/bin/sh                         system_u:object_r:bin_t:s0 ->
 system_u:object_r:shell_exec_t:s0
/sbin/agetty                    system_u:object_r:getty_exec_t:s0
/sbin/init                      system_u:object_r:bin_t:s0 ->
 system_u:object_r:init_exec_t:s0
/usr/sbin/sshd                  system_u:object_r:sshd_exec_t:s0
```

5. If the system is not working properly, check the log files in `/var/log/audit/audit.log`.

4 SELinux policy overview

The policy is the key component of SELinux. In SELinux, Mandatory Access Control (MAC) is implemented through security labels known as SELinux contexts. Each process and system resource is assigned a unique SELinux security context, also referred to as an SELinux label. An SELinux context is an identifier that focuses only on the security properties of the entity. This provides a consistent and unambiguous method for referencing objects within the SELinux policy.

An SELinux policy contains a huge number of security contexts and access rules between those security contexts. These are split into modules. This allows the administrator to customize policies for different parts of the system.

4.1 SELinux security context

The security context is a set of information assigned to a file or a process. It consists of SELinux user, role, type, level and category. This information is used to make access control decisions.

A SELinux context looks like this:

```
user_u:role_r:type_t:level:category
```

Tip: Focus on the SELinux type

For beginners working with SELinux, it is recommended to focus first on the SELinux type, which is `type_t`. This is because SELinux rules mainly use the SELinux type to define allowed access, a core concept known as *Type Enforcement*.

An understanding of SELinux types is critical, as they are frequently modified during SELinux administration.

SELINUX CONTEXT FIELDS

SELinux user

An identity defined in the policy that is authorized for a specific set of roles and for a specific *level* range. Each Linux user is mapped to only one SELinux user. However, one SELinux user can have several roles.

SELinux does not use the list of user accounts maintained by Linux in `/etc/passwd`, but uses its own database and mapping. By convention, the identity name is suffixed with `_u`, for example, `user_u`.

When a new Linux account is created and the SELinux user is not assigned to the account, the default SELinux user is used. Usually, the default value is `unconfined_u`. For a procedure on how to change the default value, refer to *Section 9.5.2, “The* **`semanage login`** *command”*.

role

Defines a set of permissions that a user can be granted. A role defines which *types* a user assigned to this role can access. By convention, the role name is suffixed with `_r`, for example, `system_r`.

type

The type conveys information about how particular files and processes can interact. SELinux policy rules define which types are allowed to access each other. This concept is called *Type Enforcement*. In the context of processes, this is also sometimes called a *domain*. By convention, the type name is suffixed with `_t`, for example, `var_t`.

level

An optional attribute that specifies the range of levels of clearance in the multilevel security.

category

An optional attribute that allows you to add categories to processes, files and users. A user can then access files that have the same category.

To check the current security context of a file, you can list the file with **`ls -Z`**. The following example lists the security context of the file `/etc/shadow`:

```
# ls -Z /etc/shadow
system_u:object_r:shadow_t:s0 /etc/shadow
```

To check the current security context of a process, you can display the current process state with **`ps -Z`**. The following example lists the security context of the `systemd` process:

```
# ps -eZ | grep systemd
system_u:system_r:init_t:s0           1 ?        00:00:01 systemd
...
```

Tip: The **`-Z`** flag convention

Many command-line tools, such as **`ls`**, **`ps`**, **`mkdir`**, **`id`**, **`cp`**, and many more, implement their explicit SELinux handling by adding a **`-Z`** flag or option. To check for explicit SELinux handling, refer to the man page of the tool and look out for **`-Z`**.

4.2 SELinux Rules

SELinux rules define whether a subject's security context (source) is permitted or denied access to an object's security context (target), explicitly specifying the allowed or denied permissions and the class of the object being accessed.

SELinux rules generally follow this format:

```
rule_name source target:class permissions;
```

rule_name

An access vector rule (AVC rule) is a rule that defines access control for object classes. The permissions for an object class are represented by a bitmask called an access vector, hence the term `access vector rule`.

source

The SELinux context of the process (subject)

target

The SELinux context of the resource (object)

class

Type of object, such as a file, process or socket

permissions

Allowed actions such as read, write and execute

Tip: Listing SELinux rules

The **`setools-console`** package contains **`sesearch`**, which is a convenient tool to list all rules.

For example, to check if the loaded policy has a rule that allows passwd (process running with `passwd_t` type) write access to the `/etc/shadow` file (labeled as `shadow_t`), run:

```
#  sesearch -A -s passwd_t -t shadow_t -p write
allow passwd_t shadow_t:file { append create getattr ioctl link lock map open read
 relabelfrom relabelto rename setattr unlink watch watch_reads write };
```

This output shows that there is a rule, so the access is allowed.

4.3 Design principles of the targeted SELinux policy

In the SELinux ecosystem, various policy types, such as the widely used targeted policy, are designed with distinct security concepts and enforcement methodologies to address diverse requirements.

SUSE Linux Micro 6.2 is shipped with the `targeted` SELinux policy by default.

The design of this policy prioritizes the confinement of long-running and network-facing services, particularly on general-purpose server systems.

While the default targeted policy is designed to cover the most common use cases in default configuration, specialized workloads often need policy adjustments. For instance, scenarios involving custom network ports or unique file system layouts will require administrators to refine the existing policy.

Additionally, SELinux Booleans encapsulate specific policy modifications that, while potentially reducing the baseline security of a general-purpose system, are necessary for certain common workloads. This design allows administrators to selectively enable these adjustments only when required, maintaining a more secure default posture.

To facilitate such modifications, SELinux offers various tools. System administrators and software developers can customize the default policy to their specific requirements.

5 Understanding SELinux logs

When SELinux blocks an action, it generates a detailed log message. Understanding these logs is very important for troubleshooting and auditing the system's security with SELinux.

The primary service responsible for handling SELinux logs is the audit daemon, **`auditd`**. On a standard system, this service is enabled by default.

- **With `auditd` running (default):** All SELinux log messages are written to `/var/log/audit/audit.log`.
- **Without `auditd` running:** If the **`auditd`** service fails or is not running, SELinux log messages can be found instead in the *systemd journal* as a fallback.

The most common log messages that are created by SELinux are **Access Vector Cache (AVC) denials**, which record that a process was denied permission to access an object.

An AVC denial is a symptom, not the problem itself, so it is important to understand why the denial occurred. When initially setting up a workload or on upgrade scenarios, SELinux blocking actions can be easily fixed by using the tools mentioned in *Section 7, "SELinux administration"*. In contrast, AVCs occurring at later stages when the workload is running for while, might indicate a blocked intrusion attempt.

5.1 Reading the audit log

The `audit.log` file contains structured records. The **`ausearch`** utility is designed to query these logs effectively. The most common use is to search for recent AVC denials.

```
# ausearch -m AVC -ts recent
type=AVC msg=audit(1753703741.779:1779): avc:  denied  { getattr } for  pid=22212
 comm="httpd-prefork" path="/srv/wwwcustom/vhosts/example.com/index.html" dev="vda3"
 ino=278 scontext=system_u:system_r:httpd_t:s0 tcontext=system_u:object_r:var_t:s0
 tclass=file permissive=0
```

Here, `-m AVC` filters for SELinux denials, and `-ts recent` limits the search to events that happened in the last 10 minutes. You can also use values like `today`, a specific date or `boot`.

The log entry contains all the information needed to understand the denial. The key fields are:

denied { getattr }

: The specific **permission** that was denied.

comm="httpd-prefork"

: The **command** that triggered the denial.

path="/srv/wwwcustom/vhosts/example.com/index.html"

: The **path** that was accessed (only for files).

scontext=...:httpd_t:...

: The **Source Context** of the process that attempted the action.

tcontext=...:var_t:...

: The **Target Context** of the object (file, directory, port) that was being accessed.

tclass=file

: The **Target Class**, which specifies the type of object (e.g., `file`, `dir`, `tcp_socket`).

permissive=0

: The **Mode** of the source domain. `permissive=0` means the security policy is enforced, while `permissive=1` means logging only.

The example log states the process named **`httpd-prefork`** (running with the `httpd_t` context) was denied permission to `getattr` a `file` that has the `var_t` context.

This example AVC occurred when a Web server was configured to use a different directory (`/srv/wwwcustom`) for the served files instead of the default path. This needs to be adjusted in SUSE Enterprise Storage as well, which can be seen in *Section 7.2, “Managing custom file contexts”*.

6 SELinux packages

SUSE Linux Micro 6.2 provides you with tools to manage SELinux on your system.

The `policycoreutils` package contains the basic tools needed for SELinux administration and is installed by default. It contains **`sestatus`**, **`restorecon`**, **`semodule`**, **`setsebool`** and many other tools.

Additionally, the `policycoreutils-python-utils` package provides helper tools that make SELinux administration easier. This includes tools such as **`audit2allow`** and **`semanage`**. If the package is missing, run the following command to install it:

```
> sudo transactional-update pkg install policycoreutils-python-utils
```

Advanced policy development requires additional `-devel` packages, which can be installed by:

```
> sudo transactional-update pkg install selinux-policy-devel policycoreutils-devel
```

7 SELinux administration

As software developers and system administrators, you may need to adapt various aspects of the SELinux policy for specialized workloads and deployment scenarios. This is because of the design principles of the *targeted* policy, as seen in *Section 4.3, “Design principles of the targeted SELinux policy”*.

Necessary adaptions might include:

- Controlling policy behavior via Booleans
- Defining and modifying file contexts
- Adjusting network port contexts
- Setting domains to `permissive` mode for troubleshooting
- Managing policy modules
- Enabling or disabling `dontaudit` rules to enhance policy development or auditing

7.1 SELinux Booleans

Instead of requiring the development and compilation of custom policy rules, booleans encapsulate predefined policy adjustments. Their primary purpose is to accommodate common operational workloads that, if permitted by default in the main policy, could potentially reduce the baseline security posture of a general-purpose system.

Instead of requiring the development and compilation of custom policy rules, booleans encapsulate predefined policy adjustments. This allows selectively enabling or disabling specific functionalities on an as-needed basis.

These changes can be applied at runtime and take effect immediately. Administrators can configure Booleans to be effective temporarily for the current session or persistently across reboots.

7.1.1 Working with Booleans

7.1.1.1 Listing Booleans

You can use the **`getsebool`** or **`semanage`** command to list currently defined Booleans. To list all currently defined Booleans, along with their state, run the following command:

```
# getsebool -a

abrt_anon_write --> off
abrt_handle_event --> off
abrt_upload_watch_anon_write --> on
...
```

To get more details about particular Booleans, you can use the **`semanage`** command as follows:

```
# semanage boolean -l

SELinux boolean              State   Default     Description

abrt_anon_write              (off  ,  off)       Allow abrt to anon write
abrt_handle_event            (off  ,  off)       Allow abrt to handle event
abrt_upload_watch_anon_write (on   ,   on)       Allow abrt to upload watch anon write
```

Additionally, confined services have a man page, which contains the corresponding Booleans and a description. Man pages have the prefix of the service name and the *_selinux* suffix: **`SERVICE_selinux`**. For example, the SELinux man page for Web servers is named **`httpd_selinux`**.

To get the status of an individual Boolean, you can use the following command:

```
# getsebool BOOLEAN_NAME
```

Alternatively, you can just use the **`grep`** command on the **`semanage boolean`** output:

```
# semanage boolean -l | grep BOOLEAN_NAME
```

A single Boolean can enable or disable several policy rules. To see which policy rules are enabled or disabled by specific Booleans, use the **`sedispol`** tool to analyze the policy file:

```
# sedispol /etc/selinux/targeted/policy/policy.34
```

As the policy rules are usually huge, we recommend setting an output file by selecting the `f` and specifying a file name. After specifying the file name, press `6`. Then you can inspect the file.

7.1.1.2 Toggling Booleans

The commands **`setsebool`** and **`semanage`** can be used to toggle the value of Booleans. You can change the Boolean status persistently or just temporarily until the session ends. To change a Boolean value temporarily, run the following command:

```
# setsebool BOOLEAN_NAME BOOLEAN_VALUE
```

where *`BOOLEAN_VALUE`* is either `on` or `off`.

To change a Boolean value persistently, run one of the following two commands:

```
# setsebool -P BOOLEAN_NAME BOOLEAN_VALUE
```

Alternatively, using the **`semanage`** command:

```
# semanage boolean -m --BOOLEAN_VALUE  BOOLEAN_NAME
```

where *`BOOLEAN_VALUE`* is either `on` or `off`.

7.2 Managing custom file contexts

When changing the default location of configuration files or other files of a confined service, you likely need to define or modify the security contexts. These contexts, also called *labels*, are attached to certain files by SELinux.

For instance, if a Web server is configured to serve Web pages from a custom directory other than its default location, the corresponding SELinux file contexts need to be updated as well to permit this operation.

To update the security context of the files in a custom location permanently, you can use the **`semanage fcontext`** command. This command writes the new file context to the policy. Additionally, the **`restorecon`** command needs to be run afterward to reflect the policy change in the file system.

The challenge when updating the SELinux context for a workload is determining which SELinux context is needed during the customization. You can list all mappings of security contexts to files in the file system using:

```
# semanage fcontext -l
```

This will show a long list of regular expressions, to which file type they match and which security context the files will be assigned. This can be searched to find the appropriate labels.

EXAMPLE 1: VIEWING DEFAULT FILE CONTEXTS

```
# semanage fcontext -l | less
SELinux fcontext                                   type               Context

/                                                  directory
 system_u:object_r:root_t:s0
/.*                                                all files
 system_u:object_r:default_t:s0
/[^/]+                                             regular file
 system_u:object_r:etc_runtime_t:s0
/\.autofsck                                        regular file
 system_u:object_r:etc_runtime_t:s0
/\.autorelabel                                     regular file
 system_u:object_r:etc_runtime_t:s0
/\.journal                                         all files          X:>>None>>
/\.suspended                                       regular file
 system_u:object_r:etc_runtime_t:s0
/a?quota\.(user|group)                             regular file
 system_u:object_r:quota_db_t:s0
/afs                                               directory
 system_u:object_r:mnt_t:s0
/bin                                               directory
 system_u:object_r:bin_t:s0
/bin/.*                                            all files
 system_u:object_r:bin_t:s0
```

Another way is to consult the SELinux man page of the service that has the custom path and check which type needs to be set. The man page convention is **SERVICE_selinux**. For example, the SELinux man page for Web servers is named **httpd_selinux**.

When the correct type for the custom path has been identified, you can set it by using **semanage fcontext** and using the TYPE and a regular expression REGEX to match the custom file path:

```
 # semanage fcontext -a -t TYPE REGEX
```

For example, to set all files below /srv/wwwcustom to have the type httpd_sys_content_t:

```
 # semanage fcontext -a -t httpd_sys_content_t "/srv/wwwcustom(/.*)?"
```

You can verify if your modification is set up correctly by using the **matchpathcon** command:

```
# matchpathcon /srv/wwwcustom
/srv/wwwcustom system_u:object_r:httpd_sys_content_t:s0
```

Then reflect the security context in the file system for that directory by resetting the security context:

```
# restorecon -Rv /srv/wwwcustom
```

Tip: Set equivalencies between the label of the default and custom location

When you are using a different location in the file system than the default one, you can check which security context would have been assigned to the default location and assign the same to the new location. For example, the default served location of a Web server is /srv/www. You want to move it to /srv/wwwcustom. Check out the label of the default location:

```
# matchpathcon /srv/www
/srv/www system_u:object_r:httpd_sys_content_t:s0
```

Check out the label of the new custom location:

```
# matchpathcon /srv/wwwcustom
/srv/wwwcustom system_u:object_r:var_t:s0
```

Set an equivalency between /srv/www and /srv/wwwcustom by running:

```
# semanage fcontext -a -e /srv/www /srv/wwwcustom
```

The new custom location now has the same label as the default one:

```
# matchpathcon /srv/wwwcustom
/srv/wwwcustom system_u:object_r:httpd_sys_content_t:s0
```

Then, reset the security contexts in the file system:

```
# restorecon -Rv /srv/wwwcustom
```

Your workload should work now as expected.

In case you would like to remove the customization again, run:

```
# semanage fcontext -d "/srv/wwwcustom(/.*)?"
```

7.3 Managing SELinux port contexts

By default, an SELinux-confined service can only bind to network ports that are defined in the policy with the correct port type. If you configure a service to listen on a non-standard port, SELinux will prevent it from starting.

For example, if you configure the SSH daemon to listen on port 2222, SELinux blocks it because only ports with the `ssh_port_t` type are accessible to the SSH service. To allow this, you must add the new port number to the SELinux policy.

To add a custom port definition permanently, you can use the **`semanage port`** command. This command writes the new port definition directly into the active SELinux policy. Unlike file contexts, you do not need to run a separate command like the **`restorecon`** command; the changes take effect immediately. However, you may need to restart the affected service.

To identify the correct port type for your service, list all existing port definitions:

```
# semanage port -l
```

This shows a list of port types, protocols and the port numbers or ranges assigned to them. You can search this list or consult the service's SELinux man page. For example, use the **`man sshd_selinux`** command to find the appropriate port type.

EXAMPLE 2: VIEWING DEFAULT PORT CONTEXTS

```
# semanage port -l | less
SELinux Port                          Proto    Port Number

...
ssh_port_t                     tcp      22
...
```

Once you have identified the correct port type, you can add your custom port using **`semanage port`** with the port type `TYPE`, protocol `PROTO`, and port number `PORT`:

```
# semanage port -a -t TYPE -p PROTO PORT
```

For example, to allow the SSH daemon to bind to the custom TCP port 2222, use:

```
# semanage port -a -t ssh_port_t -p tcp 2222
```

You can then verify your modification was added to the policy by listing the ports again and filtering for the type:

```
# semanage port -l | grep ssh_port_t
ssh_port_t                     tcp      2222, 22
```

After adding the port definition and modifying the port on the actual SSH daemon side, restart the SSH service and the daemon should be accessible on the custom port 2222.

If you need to remove the customization, use the -dflag:

```
# semanage port -d -t ssh_port_t -p tcp 2222
```

7.4 Setting permissive domains

When troubleshooting an issue with a specific service, it can be useful to disable SELinux enforcement for that specific service. Making a domain permissive means that SELinux logs policy violations for that domain but does not block any actions. This is a more targeted and secure alternative to switching the entire system to `permissive` mode with **`setenforce 0`** command.

The **`semanage permissive`** command allows you to manage a list of domains that operate permissively even while the system is enforcing the policy globally. This change is persistent across reboots.

7.4.1 Finding the appropriate domain

Before you can make a service permissive, you must find the correct domain name associated with its process. You can do this by listing running processes with their security contexts using the **`ps -eZ`** command and filtering for the process name.

For example, to find the domain for the HAProxy service:

```
# ps -eZ | grep haproxy
system_u:system_r:haproxy_t:s0    45248 ?        00:00:00 haproxy
```

The first column of the output shows the context, and the type component (ending in `_t`) is the domain. In this example, the domain for HAProxy is `haproxy_t`.

7.4.2 Setting a domain to permissive mode

Once you know the domain, you can add it to the permissive list using the **`semanage permissive -a`** command. This command makes the specific domain permissive without affecting the global enforcing mode.

```
# semanage permissive -a haproxy_t
```

Verify that the domain has been added to the list of custom permissive domains:

```
# semanage permissive -l
```

```
Builtin Permissive Types
...

Customized Permissive Types

haproxy_t
```

With the `haproxy_t` domain now permissive, the HAProxy service can perform any action, even those that violate the SELinux policy. Instead of being blocked, these violations are recorded as AVC denials in the audit log `/var/log/audit/audit.log` with a `permissive=1` flag. This allows you to gather data on what rules might be missing from your policy without interrupting the service's operation.

After you have finished troubleshooting, you must return the domain to `enforcing` mode to ensure the system is secure. To do this, delete the domain from the permissive list using the `-d` flag:

```
# semanage permissive -d haproxy_t
```

The `haproxy_t` domain is now fully protected by the SELinux policy again.

7.5 Working with SELinux modules

The **`semanage module`** command is used to manage policy modules. Some non-SELinux packages ship their independent policy modules, which enhance the rules in the main SELinux policy. These are usually installed only when the application package is installed.

Additionally, system administrators and software developers can enhance the SELinux policy with their custom set of rules, by introducing a custom policy module.

To view all modules currently installed in the policy, along with their priority and language, use the `-l` option. Modules included in the main policy usually have priority 100, whereas a priority of 200 suggests the module is included in a different package. Custom modules added by the user have a priority of 400 if not specified differently, for example:

```
# semanage module -l

Module Name               Priority  Language

abrt                      100       pp
accountsd                 100       pp
acct                      100       pp
afs                       100       pp
...
```

To disable a module for troubleshooting, use the `-d` option:

```
# semanage module -d MODULE_NAME
```

For example, to disable the **`container`** module, run:

```
# semanage module -d container
```

To verify that the module has been disabled, check the output of **`semodule -l`**. It should add a `Disabled` column:

```
# semanage module -l | grep container
container                  200       pp     Disabled
```

To reenable a module, run:

```
# semanage module -e MODULE_NAME
```

To permanently remove a module and all of its rules from the policy store, use the `-r` flag. This option is usually only needed when working with custom SELinux modules created by the user. It is not recommended or supported to remove modules that were shipped with the operating system.

```
# semanage module -r my-local-module
```

7.5.1 Creating a custom module (advanced)

If the diagnosis indicates a rule is genuinely missing from the policy and there is no policy module available for the specific workload, you can generate a custom module. Many open source projects provide their own SELinux module, which can be used to enhance the existing policy, so check for the specific workload. This workflow includes a critical review step.

1. Use the **`ausearch`** command to fetch all the AVCs in the time frame. Then, optionally, isolate the specific denials related to your issue with **`grep`**. Pipe the results to **`audit2allow`**. For example, to generate a module only for denials that have occurred recently, use:

   ```
   # ausearch -m avc -ts recent | audit2allow -M my-fix
   ```

   This command creates two files: `my-fix.te` (human-readable source) and `my-fix.pp` (installable module).

2. Critically review the contents of the human-readable `.te` file to ensure the proposed rules are appropriate and expected.

```
> cat my-fix.te
```

Warning: Security analysis of generated rules

The **audit2allow** tool translates denials into allow rules mechanically. It does not understand context or intent. Only proceed with installing a module if you fully understand the rule and are certain it does not violate your security principles.

3. *Optional*. Modify and rebuild the module: If the generated rules need to be adjusted, edit the `.te` file. Save your changes and you can then manually recompile the module.

   1. Use a text editor to modify `my-fix.te`.

   2. Recreate `my-fix.pp`:

      ```
      # make -f /usr/share/selinux/devel/Makefile my-fix.pp
      ```

4. After a review, the generated rules are deemed safe and necessary for a legitimate application function. Install the compiled module package.

   ```
   # semodule -i my-fix.pp
   ```

8 SELinux troubleshooting

An SELinux issue occurs when the security policy blocks a legitimate action that an application needs to perform its function. This topic provides a systematic workflow for resolving these issues. The process involves confirming that SELinux is the cause, gathering evidence from logs, diagnosing the underlying problem and implementing the correct solution.

8.1 Confirm the issue is SELinux related

The initial diagnostic step is to determine if SELinux is the cause of the unexpected behavior. This is done by temporarily placing the system into **permissive mode**, where denials are logged but not enforced.

1. Switch to permissive mode:

   ```
   # setenforce 0
   ```

2. Retry the action that was previously failing.

3. Analyze the result:

   - If the application **succeeds**, the problem is confirmed to be related to the SELinux policy. Proceed to the next step.

   - If the application **still fails**, the root cause is likely unrelated to SELinux and may involve other factors such as standard file permissions, firewall rules, or application configuration.

Important

It is important to return the system back to enforcing mode once the test is completed:

```
# setenforce 1
```

8.2 Gather evidence to find the denial

Once SELinux is confirmed as the cause, the next step is to find the specific denial message in the audit logs. The primary log file is located in `/var/log/audit/audit.log`. Use the **`ausearch`** utility to query these logs for recent Access Vector Cache (AVC) denials.

```
# ausearch -m avc -ts recent
 type=AVC msg=audit(1753703741.779:1779): avc:  denied  { getattr } for  pid=22212
 comm="httpd-prefork" path="/srv/wwwcustom/vhosts/example.com/index.html" dev="vda3"
 ino=278 scontext=system_u:system_r:httpd_t:s0 tcontext=system_u:object_r:var_t:s0
 tclass=file permissive=0
```

This log entry contains the essential details: the source process context (`scontext`), the target object's context (`tcontext`) and the permission denied. More information on interpreting the log file can be found in *Section 5, “Understanding SELinux logs”*.

8.3 Understanding the cause of the denial

After reviewing the log, the denial is likely caused by one of the following common issues:

Incorrect file label

This denial typically occurs in one of two scenarios: a file has an incorrect label for its current location or the file is in a non-standard location and needs to be assigned a correct label in the policy.

Service using a non-standard port

A denial occurs if a service attempts to bind to a network port that is not assigned the correct port type in the SELinux policy.

Missing policy rule

The system's SELinux policy may lack a specific rule required for an application's operation. This happens when the application is not shipped by default with the distribution. In this case, the resulting AVC denial is valid and indicates the policy needs to be extended.

8.4 Applying the correct fix

Based on the diagnosis, apply the appropriate solution.

For incorrect file labels

The solution depends on the cause of the mislabeling:

- **Case 1: Incorrectly labeled file in the file system.** If files in a standard location have the wrong labels, for example, after using **`mv`** or **`cp`** without the **`-Z`** flag, use the **`restorecon`** command to reset them to the policy default. For example, to restore the context of `/srv/www/vhosts/example.com`, use:

  ```
  # restorecon -Rv /srv/www/vhosts/example.com
  ```

- **Case 2: Tell SELinux about a custom path.** If you intentionally chose a non-standard location for your files, for example, serving Web content from `/srv/wwwcustom`, you must add a new rule to the SELinux policy with the**`semanage fcontext`**command. Then, use the **`restorecon`** command to apply the new rule.

  ```
  # semanage fcontext -a -t httpd_sys_content_t "/srv/wwwcustom(/.*)?"
  # restorecon -Rv /srv/wwwcustom
  ```

  For more information on adding custom file paths, see *Section 7.2, “Managing custom file contexts”*.

For non-standard ports

Use the **`semanage port`** command to add the custom port to the policy with the correct type. For example:

```
# semanage port -a -t ssh_port_t -p tcp 2222
```

For more information on adding custom file paths, see *Section 7.3, “Managing SELinux port contexts”*.

For missing policy rules

If the issue is that policy rules are missing, ensure to check if the corresponding module is loaded and enabled:

```
# semanage module -l | less
```

If that is the case, or there is no corresponding module as the workload is highly customized, you can consider writing and including a custom policy module with your own rules as described in *Section 7.5.1, “Creating a custom module (advanced)”*.
To unblock the workload while working on the custom module, you can set the domains blocking the access into `permissive` mode, as described in *Section 7.4, “Setting permissive domains”*.

8.5 Advanced Scenarios and Recovery

8.5.1 Handling missing log entries

If an application fails but no denial is logged, a `dontaudit` rule may be silently suppressing it. You can temporarily disable these rules to make all denials visible.

1. Disable all `dontaudit` rules:

   ```
   # semodule -DB
   ```

2. Retry the failing action to generate the denial log.

3. Reenable the rules to restore normal logging behavior:

   ```
   # semanage -B
   ```

8.5.2 Recovering from an SELinux-related boot failure

If an SELinux policy or labeling issue prevents the system from booting, the primary goal is to regain access to a shell to diagnose and fix the problem. This is achieved by booting the system in `permissive` mode.

1. Restart the machine and interrupt the boot sequence to access the GRUB menu.

2. Select the desired kernel and press `e` to edit its boot parameters.

3. Locate the line beginning with the keyword `linux`.
4. Append the parameter `enforcing=0` to the end of this line. This instructs the kernel to boot in `permissive` mode for this session only.
5. Press Ctrl–X or F10 to proceed with booting.
6. Once the system has booted successfully, log in and use the standard troubleshooting tools described in this chapter to find the denial and apply the correct fix.

Note: Fixing widespread labeling issues at boot

In the specific case where the boot failure is caused by incorrect file labels across the entire system, the most effective solution is to schedule a full relabel. After booting into `permissive` mode using the steps above, execute the following commands before rebooting:

```
# touch /etc/selinux/.autorelabel
# reboot
```

The system now reboots and begins a comprehensive relabeling of the file system.

9 Tools for managing SELinux

The topic provides a list of tools that you can use to configure SELinux.

SUSE Linux Micro provides you with tools to manage SELinux on your system. If the below described tools are not installed on your system, install the tools by running:

```
# transactional-update pkg install policycoreutils-python-utils
```

After successful installation, reboot the system.

9.1 Using the Z option

Where SELinux is installed and configured, you can use the `-Z` to regular commands like **`ls`**, **`id`** or **`ps`**. Using this option, you can display the security context of files or processes. For example, with the **`ls`** command:

```
> ls -Z /etc/shadow
      system_u:object_r:shadow_t:s0 /etc/shadow
```

9.2 The **chcon** command

The command name **`chcon`** stands for change context. The command can change the full security context of a file to the value provided on the CLI, or it can change parts of the context. Alternatively, you can provide a file that serves as a reference. The change is not reflected in the policy, so a relabel of the file will reset the context back to its definition in the policy.

To change the full security context of a file, the command syntax looks as follows:

```
# chcon SECURITY_CONTEXT FILENAME
```

where:

- *`SECURITY_CONTEXT`* is in the format: *`SELinux_USER`*:*`ROLE`*:*`TYPE`*:*`LEVEL`*:*`CATEGORY`*. For example, the context could be: *system_u:object_r:httpd_config_t:s0*.
- *`FILENAME`* is a path to the file whose context should be changed.

To set a security context according to a provided file that serves as a reference, run **`chcon`** as follows:

```
# chcon --reference=REFERENCE_FILE FILENAME
```

where:

- *`REFERENCE_FILE`* is a path to a file that should be used as a reference.
- *`FILENAME`* is a path to the file whose context should be changed.

Alternatively, you can change only one part of the security context. The general syntax of the **`chcon`** command is as follows:

```
# chcon CONTEXT_OPTION CONTEXT_PART FILENAME
```

The options and arguments have the following meaning:

- depending on the context part, *`CONTEXT_OPTION`* can be any of the following:

  `-u` resp `--user`

  denotes that an SELinux user context will be changed on the provided file:

  ```
  # chcon -u system_u logind.conf
  ```

  `-r` resp `--role`

  only the role part will be changed in the context of the provided file:

  ```
  # chcon -r object_r logind.conf
  ```

`-t` resp `--type`

only the type part will be changed in the context of the provided file:

```
# chcon -t etc_t logind.conf
```

`-l` resp `--range`

only the range part of the security context will be changed:

```
# chcon -l s0 logind.conf
```

- *`CONTEXT_PART`* is the particular value of the security context to be set.
- *`FILENAME`* is a path to the file whose context will be changed.

Note: Using **`chcon`** on symbolic links

By default, when you change the security context on a symbolic link, the context of the link target is changed and the symbolic link context is **not** changed. To force **`chcon`** to change the context of the symbolic link and not the link target, use the `--no-dereference` option as shown below:

```
# chcon --no-dereference -u system_u -t etc_t network.conf
```

You can change the context of all files in a directory by using the recursive option:

```
# chcon --recursive  system_u:object_r:httpd_config_t:s0 conf.d
```

9.3 **`getenforce`** and **`setenforce`** commands

The **`getenforce`** command returns the current SELinux mode: `Enforcing`, `Permissive` or `Disabled`.

```
# getenforce

Permissive
```

The **`setenforce`** command temporarily changes the SELinux mode to enforcing or permissive. You cannot use this command to disable SELinux. Remember that the change persists only until the next reboot. To change the state permanently, follow the description in *Section 3.1, “Changing the SELinux mode”*.

```
# setenforce MODE_ID
```

where *MODE_ID* is *0* for the `permissive` mode or *1* for the `enforcing` mode.

9.4 The **fixfiles** script

The script enables you to perform the following tasks with the security context:

- check if the context is correct
- change any incorrect file context labels
- relabel your system if you added a new policy

The script syntax is as follows:

```
# fixfiles [OPTIONS] ARGUMENT
```

where:

- *OPTIONS* can be the following:

  `-l` *LOGFILE*

  saves the output to the provided file

  `-o` *OUTPUT_FILE*

  saves to the provided output file the names of all files whose file context differs from the default

  `-F`

  forces a reset of context

- *ARGUMENT* can be one of the following:

  check

  shows previous and current file context for an incorrect label without performing any changes

  relabel

  relabels incorrect file contexts according to the currently loaded policy

  restore

  restores incorrect file contexts to the default values

  verify

  lists all files with incorrect file context labels without performing any changes

9.5 The **semanage** command

The **semanage** command can be used to configure parts of the policy without the need to recompile the policy from sources. The command enables you to perform the following tasks:

- manage Booleans by using the `boolean` argument. For details about Booleans, refer to *Section 7.1.1, "Working with Booleans"*.
- adjust the context of files by using the `fcontext` argument.
- manage user mappings using the `login` argument
- manage SELinux users using the `user` argument
- manage SELinux policy modules using the `module` argument

The general command syntax looks as follows:

```
# semanage ARGUMENT OPTIONS [OBJECT_NAME]
```

where:

- *ARGUMENT* is one of the following: **login**, **user**, **fcontext**, **boolean**, **module**.
- *OPTIONS* depends on the provided *ARGUMENT*. Common options are described in *Common options*.
- *OBJECT_NAME*, depending on the provided *ARGUMENT*, can be a login name, module name, file name or SELinux user.

COMMON OPTIONS

`-a`, `--add`

adds a provided object

`-h`, `--help`

prints the command help

--extract

displays commands that were used to change the system (Booleans, file context, and so on)

`-l`, `--list`

lists all objects

`-m`, `--modify`

modifies the provided object

`-n`, `--noheading`

modifies the output of the listing operation by omitting headings

`-s`, `--seuser`

specifies the SELinux user

Other options are specific to particular **`semanage`** commands and are described in corresponding sections.

9.5.1 The `semanage fcontext` command

Using the **`semanage fcontext`** command, you can perform the following tasks:

- query file context definitions
- add contexts on files
- add your own rules

Changes performed to the file context using the **`semanage fcontext`** command do not require modifications or recompilation of the policy.

On top of the common options described in *Common options*, the **`semanage fcontext`** command takes the following options:

`-e`, `--equal`

The option enables you to use the context of the provided path context to label files in a different directory (the provided target path). For example, you want to assign the same context as `/home` has to an alternative home directory `/export/home`. If you use this option, you need to provide the source path and the target path:

```
# semanage fcontext -a -e /home /export/home
```

`-f`, `--ftype`

To specify a file type. Use one of the following values:

- `a` - all files, which is also the default value
- `b` - a block device
- `c` - a character device
- `d` - a directory
- `f` - regular files

- `l` - a symbolic link
- `p` - a named pipe
- `s` - a socket

9.5.2 The `semanage login` command

The **`semanage login`** enables you the perform the following tasks:

- Mapping of Linux users on a particular SELinux user. For example, to map the Linux user *tux* on `sysadm_u`, run the command:

  ```
  # semanage login -a -s sysadm_u tux
  ```

- Mapping of a group of Linux users on a particular SELinux user. For example, to map users of the *writers* group on `user_u`, run the command:

  ```
  # semanage login -a -s user_u %writers
  ```

  The group is then listed in the output of **`semanage login -l`**, prefixed with the % character. Keep in mind that the user group should be primary because mapping SELinux users on supplementary groups may result in incompatible mappings.

  ```
  # semanage login -m -s staff_u %writers
  ```

- Mapping of Linux users on a particular SELinux MLS/MCS security range.
- Modifying of the already created mapping. For this purpose, just replace the `-a` option with `-m` in the previous commands.
- Setting the default SELinux user for new Linux users. The usual default SELinux user is `unconfined_u`. To change the value to `staff_u`, run the command:

  ```
  # semanage login -m -s staff_u __default__
  ```

9.5.3 The `semanage boolean` command

The **`semanage boolean`** command is used to control Booleans in the SELinux policy.

The command synopsis looks as follows:

```
semanage boolean [-h] [-n]  [ --extract |
```

```
        --deleteall | --list [-C] | --modify ( --on | --off | -1 | -0 ) boolean ]
```

On top of the common options, you can use the following ones specific to the **semanage boolean** command:

--list -C

To display a list of local modifications to Booleans.

-m --on | -1

To switch the provided Boolean on.

-m --off | -0

To switch the provided Boolean off.

-D, --deleteall

To delete all local modifications to Booleans.

The most common usage of the command is to switch on or off a particular Boolean. For example, to switch on the authlogin_yubikey Boolean, run:

```
# semanage boolean -m --on authlogin_yubikey
```

9.5.4 The semanage user command

The **semanage user** command controls the mapping between the SELinux user and the roles and MLS/MCS levels.

On top of the common options described in *Common options*, the **semanage user** command takes the following options:

-R [ROLES], --roles [ROLES]

A list of SELinux roles. You can enclose multiple roles within double quotes and separate them by spaces, or you can use the -R several times.

Using this command, you can perform the following tasks:

- Listing the mapping of SELinux users on roles by running:

```
# semanage user -l
```

- Changing the roles assigned to the user_u SELinux user:

```
# semanage user -m -R "system_r unconfined_r user_r"
```

- Assigning to `admin_u` the role `staff_r` and a category `s0`:

  ```
  # semanage user -a -R "staff_r -r s0 admin_u
  ```

- Creating a new SELinux user, for example, `admin_u` with the `staff_r` role. You also need to define the labeling prefix for this user by using the `-P`:

  ```
  # semanage user -a -R "staff_r" -P admin admin_u
  ```

9.5.5 The **semanage module** command

The **`semanage module`** command can install, remove, disable or enable SELinux policy modules.

On top of the common options described in *Common options*, the **`semanage fcontext`** command takes the following options:

`-d`, `--disable`

To disable the provided SELinux policy module:

```
# semanage module --disable MODULE_NAME
```

`-e`, `--enable`

To enable the provided SELinux policy module:

```
# semanage module --enable MODULE_NAME
```

9.6 The **sestatus** command

The **`sestatus`** gets the status of a system where SELinux is running.

The generic syntax of the command looks as follows:

```
sestatus [OPTION]
```

When run without any options and arguments, the command outputs the following information:

```
# sestatus

SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
```

```
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     requested (insecure)
Max kernel policy version:      33
```

The command can take the following options:

-b

Displays the status of Booleans on the system.

-v

Displays the security context of files and processes listed in the `/etc/sestatus.conf` file.

10 Disabling and reenabling SELinux

Important: Disabling SELinux is not recommended

It is not recommended to disable SELinux. If SELinux is possibly causing issues to your system, switch to the `permissive` mode to debug your system.

SELinux is enabled by default in SUSE Linux Micro 6.2. To disable or reenable SELinux requires a modification of the kernel command-line. This can be done temporarily for one boot or permanently by modifying the boot loader configuration.

10.1 Modify the kernel command-line via the GRUB boot loader configuration

To disable SELinux, edit the GRUB configuration file to include `selinux=0`:

```
# update-bootloader --add-option "selinux=0"
# update-bootloader --config
```

Verify that `/etc/default/grub` contains `selinux=0`:

```
GRUB_CMDLINE_LINUX_DEFAULT="...... selinux=0"
```

The change will be effective after reboot.

To verify that SELinux is disabled, run:

```
# sestatus
```

```
SELinux status:                 disabled
```

To enable SELinux again, edit the GRUB configuration file to remove the `selinux=0` option and relabel during reboot.

```
# update-bootloader --del-option "selinux=0"
# update-bootloader --config
# touch /etc/selinux/.autorelabel
```

SELinux is reenabled after reboot.

Note: Relabeling your system after reenabling SELinux

If you disable SELinux on your system and then enable it later, make sure that you relabel your system. When SELinux is disabled and you perform changes to your file system, the changes are not reflected in the context; for example, new files do not have any context.

You must relabel your system by using the **`restorecon`** command and the `autorelabel` boot parameter. You can also create a file that will trigger relabeling on the next boot. To create the file, run the following command:

```
# touch /etc/selinux/.autorelabel
```

After reboot, the file `/etc/selinux/.autorelabel` is replaced with another flag file, `/etc/selinux/.relabelled`, to prevent relabeling on subsequent reboots.

11 Legal Notice

Copyright© 2006–2025 SUSE LLC and contributors. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or (at your option) version 1.3; with the Invariant Section being this copyright notice and license. A copy of the license version 1.2 is included in the section entitled “GNU Free Documentation License”.

For SUSE trademarks, see https://www.suse.com/company/legal/ ↗. All other third-party trademarks are the property of their respective owners. Trademark symbols (®, ™ etc.) denote trademarks of SUSE and its affiliates. Asterisks (*) denote third-party trademarks.

All information found in this book has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Neither SUSE LLC, its affiliates, the authors, nor the translators shall be held liable for possible errors or the consequences thereof.

A GNU Free Documentation License

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the

Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See https://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
```

```
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled “GNU
Free Documentation License”.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.