Mandatory Access Control - Getting started with Tomoyo Linux on Slackware

Introduction

There are a few different tools in the Tomoyo family. Mainly Tomoyo 1, Akari and Tomoyo 2. There is also CaitSith, but **this guide is dealing with Tomoyo 2.x**. And at the time of writing Tomoyo 2.6.x for Kernel 5.1 and later.

Tomoyo Linux is very minimalistic (but complex) and in my view very much in harmony with the Slackware way of doing things. It has very few dependencies and is fairly easy to get started with. The documentation is excellent, and can easily be used to get started with Tomoyo. Tomoyo doesn't use anything Python or things like that (like SELinux), it uses command line tools and an ncurses interface (optional). So, then, why am I even bothering to write this?
The main reason is to add information specific to Slackware, but also to write down the basics from a user perspective. You can probably just use the main Tomoyo documentation to get started, but there might be a few questions after that and/or Slackware specifics to do. Please DO use the official documentation but feel free to use this in addition.

Index of the manual:
https://tomoyo.sourceforge.net/2.6/index.html.en
Section about “installing” Tomoyo: (which is basically what this guide covers)
https://tomoyo.sourceforge.net/2.6/chapter-3.html.en

So, what is the purpose of Tomoyo? The purpose is to implement **Mandatory Access Control** (MAC) on your system, which can be used in alot of ways to secure different aspects of the system. MAC works in such a way that EVERYTHING is FORBIDDEN unless you explicitly permit it (with policy). It is a fairly simple to implement MAC which can do alot for the security of your system! Among others, the manual specifically mentions SSH and Apache as some examples. If you read those you might start to understand how it can be extremely useful. (ssh example)
https://tomoyo.sourceforge.net/2.6/chapter-12.html.en
For a distro like Slackware, it is fairly easy to implement MAC for the whole boot sequence, which means the system can ONLY do what it is set to allow. That might sound impractical, and it is, and it might sound scary as well, but it is not, because **the default mode of Tomoyo is to do nothing**. Only if you enforce policies will it block everything else than what you allow, and as mentioned, making a MAC policy for booting Slackware is fairly simple, because the boot process is fairly simple. Creating a MAC policy for a regular user system is quite a big task, but Tomoyo comes with a “learning mode” which writes policies, but do not enforce them, so it is a managable task overall.

Preliminaries

If you do not know how to build your own Kernel and deal with the bootloader then do not proceed with these things, it is not necessarily dangerous, but inadvisable. First learn how to build your own Kernel AND deal with your bootloader, then proceed. Only procede on a production system if you have backed up your data and you are

confident of what you are doing.

To use Tomoyo Linux you must enable it in the kernel. The most common option is to compile your own Kernel and run it. You can copy the config file from your running Kernel and use that for your new Kernel, and just add the options for Tomoyo in the security section of the kernel, then compile. Here is an example of how you can build a Kernel:
https://docs.slackware.com/howtos:general_admin:brief_kernel_build

Choose the following options in the “Security options” section:

```
[*] Enable different security models
-*- Enable the securityfs filesystem
-*- Security hooks for pathname based access control
[*] TOMOYO Linux Support
(51200) Default maximal count for learning mode
(1024) Default maximal count for audit log
[ ]   Activate without calling userspace policy loader.
(/sbin/tomoyo-init) Location of userspace policy loader
(/sbin/init) Trigger for calling userspace policy loader
```

You might want to set the “max learning mode” to more than 2048, as some processes can create ALOT of entries. But don't set it too high, as it will consume kernel resources and slow down your boot significantly if you have millions of entries. The default is set in Tomoyo, not the Kernel, so you can set it higher in the Kernel and adjust it to whatever you actually want it to be in Tomoyo. If you have a fairly modern computer 51200 might be a useful max value. Audit value can be left as is.

You need to add your new kernel to your boot manager, lilo or grub or others, and to enable Tomoyo at boot you can add “security=tomoyo” to the boot line of your Tomoyo Kernel. You can use this following example in Grub for Bios mode:
https://docs.slackware.com/howtos:misc:lilo_to_grub_bios_mbr

Once you boot this Kernel with this option, you can verify that Tomoyo is enabled in the Kernel:

```
grep tomoyo_write_inet_network /proc/kallsyms
```

should return:

```
ffffffff8155e460 T tomoyo_write_inet_network
```

Tomoyo-tools

Once a Tomoyo Kernel is active you need to install the Tomoyo-tools. Download the tomoyo-tools from:
https://sourceforge.net/p/tomoyo/svn/HEAD/tree/tags/tomoyo-tools/ 2.6.1

Move the download to /usr/src/ or some other directory, then:

```
(as root)
cd /usr/src
tar -xvf tomoyo-tools-2.x.x.tar.gz
cd tomoyo-tools
make USRLIBDIR=/usr/lib64
make USRLIBDIR=/usr/lib64 install
```

Now you have the tomoyo-tools installed. Next you need to load some initial policy configurations for tomoyo.

```
/usr/lib64/tomoyo/init_policy
```

Note. You will find these policies in /etc/tomoyo

There are various ways to start Tomoyo, but the easiest is to just add to your Tomoyo Kernel boot line “security=tomoyo”. You can run different Kernels with and without this option if you need to boot without Tomoyo.

1st time domain and policy generation

Tomoyo Kernel is running and Tomoyo-tools are installed and init_policy have been run.

It's time to reboot, which will also generate domains. By default Tomoyo does nothing once you have it enabled, except temporarily log execution chains (domain generation). After reboot you can take steps to do things with policies.

Once you have rebooted you can run the policy editor to check and edit policies:

```
tomoyo-editpolicy
```

You will see that it has generated domain entries, related to your system boot.

This is the point where things actually starts happening. If you just stop here, Tomoyo will be active but do basically nothing and not affect much at all. The next steps will still leave Tomoyo in a kind of inactive state, but it will generate domains and up to 2048 acls/policies per domain. If you continue to use your computer like this and never look at Tomoyo again, it will generate alot of policies(some data) and eventually slow down your boot somewhat(due to loading policies on boot). It is still safe to leave the system in this state.

The best step to take at this point is to put all domains into learning mode. Exit tomoyo-editpolicy and run:

```
tomoyo-setprofile -r 1 '<kernel>'
```

-r is recursive, 1 is learning mode profile and <kernel> is the top domain. This means all domains under it will be set to learning mode. This change need to be saved:

```
tomoyo-savepolicy
```

and then reboot to create actual boot policies..

Learning mode and 1st boot with policies

You might already on your first boot notice that you get a message “ACL's for some domains are full, learning mode stopped”. And yes, that is probably the case, some processes do ALOT of things, and many of them might be very similar. This is where the work starts. Anyways, you can go to the policy editor to look:

```
tomoyo-editpolicy
```

You will be able to see everything that happened during boot (and after), each domain with ACL's inside, showing what each process is doing. Select one and press enter to look at entries. Each domain has a hard and a soft limit on the amount of ACL's it can contain. By default this is 2048 both in the Kernel and in the tomoyo-tools profile. You should have set this somewhat higher when you configured and compiled your Kernel, but the limit is still 2048 due to the current soft limit in the profile. This is your choice and preference, but if it is full, you will not be able to see everything that the domain is doing (new entries are not added beyond 2048).

This is where the work starts. Alot of entries under each domain will be almost identical. That's why most of the effort with LSM and Tomoyo is to write more general policies that covers thousands of ACL's with one entry. And this is what learning mode is for. Once you have a full “generalized” policy for a “domain” or a program if you want, you can start enforcing the policy. But hold your horses!

It's alot of work to rewrite policies, so you should probably first focus on some domains, or study what domains are doing. It's learning mode afterall. It does nothing, only add entries. It's more like a logging tool at this point, but it is up and running, and ready to be used.

This is a good time and place to learn how to use tomoyo-editpolicy and how to “trim” policies both with tomoyo-editpolicy and with tomoyo-patternize. It's a good time to read the whole Tomoyo manual to understand everything better and learn how to trim policies efficiently. It is a good time to test out things.

At this point you can safely leave it like it is, and leave the limit of entries at 2048. How to write and enforce policies for tomoyo is outside of the scope of THIS document. But a good place to start is the official tomoyo-manual. Yes I know, I told you to RTFM. But it's a good manual, and it's not super long or super heavy, nonetheless super useful.

Securityfs

Securitfs **should** be mounted automatically by the Kernel, but in my cases this has not been the case. To check if it is, check the content of /sys/kernel/security/ where it should be mounted.

```
ls -la /sys/kernel/security
```

If this folder has content, the mount works as it should. If it is empty, something went wrong. Try to add the entry to fstab:

```
nano /etc/fstab
securityfs /sys/kernel/security securityfs defaults 0 0
ctrl+x (save)
```

Then reboot and check /sys/kernel/security/ again like above. If you have the same problem as me, it will still not be there. The first/last/best solution to this is to use rc.S to get this mounted early in the boot process. It is fairly easy, and the ideal place is right after the /sys entry(almost at the top), and you can use that entry as an example as well. But **be careful when doing anything with rc.S, you could end up with an unbootable system!** You should end up with something like this:

```
nano /etc/rc.d/rc.S
```

```
# Mount securityfs if it is not already mounted:
if [ ! -d /sys/kernel -a -z "$container"]; then
 /sbin/mount -v securityfs /sys/kernel/security -n -t securityfs 2>
/dev/null
fi
```

```
ctrl+x (save)
```

Once you reboot and check /sys/kernel/security again, it should contain files and folders including “tomoyo”.

/etc.rc.d starting Tomoyo

Although the easiest and most flexible way to start Tomoyo is to add “security=tomoyo” to the boot line, you can also start tomoyo in other ways by using /etc/rc.d and rc.local or make an rc.tomoyo for example.

Below here are steps that you can take once you have some experience with using Tomoyo. These steps below have risks associated with them, and you should understand Tomoyo and those risks, and possible solutions to possible problems before implementing any of these important steps quite early on in your Tomoyo journey. But for sure AFTER you understand the basics somewhat.

Tomoyo-auditd

Using audit with Tomoyo is one of the most important and useful steps towards being able to fully use Tomoyo. It is useful in creating policy and moving from learning mode to enforcing policies.
However, precautions should be taken, in some “unfortunate” situations an audit log can fill up a disk

in a matter of seconds. Under normal circumstances, this will not happen, but it can happen, and you should know how to be able to handle that and/or dealing with logs and audit in general before you enable tomoyo-auditd. You might want to enable log rotations and filter audit logs properly, and you need to keep an eye on logs, you can't just leave them unattended. Under normal circumstances, it will be just fine, it is just a log, but don't activate it if you're unsure or do not plan to maintain it.

To start tomoyo-auditd on boot simply add it to /etc/rc.d/rc.local:

```
/usr/sbin/tomoyo-auditd
```

Audit logs can be found in /var/log/tomoyo/

```
ls -lha /var/log/tomoyo/
```

The audit logs actually start when Tomoyo/securityfs starts, so even if rc.local is at the end of the boot process, the audit logs will still contain entries from before the start of tomoyo-auditd.

Increasing learning entries

As described earlier, there is a hard limit and a soft limit to learning entries. If you followed the instructions above the max learning entries set by Kernel is 51200 per domain. The default max for Tomoyo is still 2048. For most domains 2048 is enough, but for some 2048 is not nearly enough, and even problematic in trying to develop policies. Some domains will easily take up 51200 as well, but this number does give you enough time and room to trim the domain policies.
However, caution should be taken, as ALL domains will have the limit of 51200, and if you're not developing and trimming policies, this can become an issue. You could experience significantly slower boot times. Don't increase this number if you don't plan on maintaining and trimming policies. Many GUI applications can take 51200 quite quickly, so a reasonable compromise if you don't plan on actively trimming and maintaining policies quite yet is something like 10240.
You can edit /etc/tomoyo/profile.conf to change the limit, by editing the entry starting with **1** as in profile 1 as in learning mode, and **preference** under “max_learning_entry=**10240**”:

```
nano /etc/tomoyo/profile.conf
1-PREFERENCE={ max_audit_log=1024 max_learning_entry=10240 }
```

You need to load that profile into the Kernel to make it effective:

```
tomoyo-loadpolicy -p < /etc/tomoyo/profile.conf
```

This same step can also be done in the tomoyo-editpolicy tool by pressing “W” to get to the options menu, and then pressing “P” for profile. Here you can edit a line by pressing “S” and verify with “enter” key. All these options can be further looked into in the ncurses interface by using the “?” key.

Appendage

Do give Tomoyo a try! It is fairly easy to use; it is a MAC usable in practice. You can use it to monitor activity and only locking down a few domains, or you can just give it a spin to see how it and MAC works.

But do read the manual. This here is just additional info to what is in the manual, and only covers how to install Tomoyo and get it up and running. The manual is easy to read and covers alot of information about how to use Tomoyo, necessary information to be able to use it, and in a very short and precise form: https://tomoyo.sourceforge.net/2.6/index.html.en

So, please don't rely on only THIS guide. It is not enough and the manual is much better.
Another similar guide, part 2 might be added at a later time, if relevant. About managing policies for Slackware and in general.

Sources

* Original source: https://tomoyo.sourceforge.net/documentation.html.en
* Originally written by zeebra

howtos, security, LSM, MAC, Tomoyo, author zeebra

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/howtos:security:tomoy_linux_basics_slackware

Last update: **2023/12/17 10:31 (UTC)**