Tomoyo Linux

June 2017 OLUG

by Aaron Grothe

What is Tomoyo?

From Wikipedia: Tomoyo is a feminine Japanese given name is a variant transcription of the name Tomoko. The name means wise era or worldly wisdom

What is Tomoyo Linux?

- Tomoyo Linux is a Linux Security package that provides MAC for Linux. It also can do some very cool things in terms of training, generating policies, monitoring systems, etc.

What is MAC?

To describe MAC we'll start with DAC (Discretionary Access Control). You already live with DAC. The basic idea here is that the owner of a resource. E.g. Chad can control who has access to it. In general linux terms this is owner, group, and world privs

Mac Cont'd

MAC is Mandatory Access Control it is similar to DAC except there is a concept of the security administrator. E.g. I have an Excel file. The security administrator could create a policy that would not be able to be overwrite to say whether or not I can hand access to this file to other people. I can't override it.

MAC can depending on implementation even prevent root from doing things. Thought experiment can God microwave a burrito so hot God couldn't eat it?

History of Tomoyo LInux

- Tomoyo Linux was launched in 2003 and was sponsored by NTT until 2012
- There are three distinct versions of Tomoyo to consider
  - Tomoyo 1.x - this version is a set of patches to the kernel and tools, not part of the Linux kernel source code
  - Tomoyo 2.x - not as full-featured implemented as a LSM along with (AppArmor, Smack, and SeLinux), integrated with the Linux kernel source code
  - Akari working towards bringing all the Tomoyo 1.x features to Tomoyo 2.x

Why Tomoyo Linux

Tomoyo Linux runs in four interesting modes

- Learning - you can dynamically create a policy - figure out what you webserver is and isn't allowed to do
- Disabled - won't do anything to your system allows normal operation
- Permissive - will allow all operations, but not add the requests to your policy
- Enforcing - applying a policy to the system, operations not explicitly allowed are denied. You can do this by having a policy generated, or you can code one by hand

Comparing SELinux to Tomoyo Linux

- Tomoyo has an easier syntax
- Tomoyo is pathname based like AppArmor
  - E.g. if /etc/shadow is linked to /tmp/shadow, they are different files to Tomoyo, unlike SELinux which is object based - but no need to relabel all the time
- Tomoyo was not created by the NSA
- Nobody expects Tomoyo Linux
- Theoretically you can experimentally stack LSMs so you could possibly stack Tomoyo on SELinux and run them both together on a system
- SELinux file labeling - you'll learn to hate it

What Distros work with Tomoyo?

- Debian/Ubuntu available out of the box
- Fedora/CentOS/RHEL need to recompile kernel for it
  - They only support SELinux out of the box
- Arch not default, need to recompile kernel for it - nice documentation exists for it
- Pretty much any distro that can have its kernel recompiled

Does my Distro support Tomoyo?

$ grep tomoyo_write_inet network/proc/kallsyms

If you get a response you are halfway there. Now all you need are the tools to control tomoyo.

Demo Tomoyo2.x on Debian VM

Confirm that kernel installed support Tomoyo

$ grep tomoyo_write_inet_network /proc/kallsyms

Install the Tomoyo-tools on the system

$ apt-get install tomoyo-tools

Demo cont'd

Update grub to add the tomoyo linux to the system

$ vi /boot/grub/grub.cfg

Add "security=tomoyo" to linux line

$ reboot

YOU ARE NOW GOING TO BE LIVING IN A TOMOYO WORLD!!!

Demo Cont'd

Well that didn't work :-(

WHY???

We don't have a profile.

So we'll disable tomoyo and get it up and running

> disable

Demo Cont'd

So what went wrong?

We don't have a policy. We'll make a policy

sudo /usr/lib/tomoyo/init_policy

reboot

NOW that's better

Demo Cont'd

So now we have a Tomoyo enabled system up and running

Time to fire up the policy editor

tomoyo-editpolicy /etc/tomoyo

Demo Cont'd

We'll take a look at the available profiles/modes

W/w - switches between modes

P/p - shows profiles

Profile 0 / disabled - ignores profile, all allowed
Profile 1 / learning - allows / adds to profile
Profile 2 / permissive - allows /does not add to profile
Profile 3 / enforcing mode - enforces profile

Demo Cont'd

We'll set the kernel to learning mode and reboot

N/n - Namespace editor
S/s - set the domain
1 - set to learning mode
Q/q - quit

Reboot to get the system running in learning mode

reboot

Demo Cont'd

System boots pretty normally except it is running Tomoyo Linux in the background and monitoring the access for every process

Time to fire up the policy editor and take a look

tomoyo-editpolicy

Time to see what the kernel has executed

Time to see about training a session

For this we'll be running some commands in ksh

ksh

ls - then refresh in editpolicy
whoami - then refresh in editpolicy
ping www.google.com - then refresh in editpolicy

Ok. So now we have a basic policy

Demo Cont'd

Time to drop the hammer

S/s - change profile section

3 - change to enforcing

try a couple of the commands we've already done

ls, whoami, etc

try one we haven't tried before

Demo Cont'd

Try ssh

ssh - should result in permission denied

Flip the ksh back to learning mode and run ssh, then put back into enforcing mode

#S 1 / ksh

ssh

S 3 / ksh

Demo Cont'd

Now we can run ssh

We can remove entries from the domain by using the Delete command

Now we'll pull ping from the domain

highlight ping

D / then confirm

Demo Cont'd

Time to Lock Down ping

ping www.google.com

Successful / since we're in mode 1

Set it to mode 3 enforcing

S 3

Demo Cont'd

Ping www.nebraskacert.org

ping www.nebraskacert.org

Denied. This is because of the granularity of the policy. We can modify the policy or put it back into learning mode and add this new option

Ok. Time to reboot and see how it works

Demo Cont'd

Fire up ksh and verify the current state of things

ksh

Ping, good
Telnet, bad that isn't supported

WTF - fire up the editpolicy

Demo Cont'd

None of our changes are saved?

You have to run tomoyo-savepolicy to have it save your policy, otherwise it is lost when you reboot

Demo Cont'd

Can we add a command to a profile by hand

Yes but it is a bit of a pain and is beyond the demo. You'll probably want to use learning mode and copy policies around by hand to get it working

About the only change I've made to a policy is to broaden options like ssh or file access by using wildcards

Time to take a look at /etc/tomoyo and how it is laid out

Demo Cont'd

Review the directory structure of the configs

/etc/tomoyo/policy

Lists current and previous policies, very useful for getting around the system

/etc/tomoyo

Holds various conf files for the system

Demo Cont'd

So we've gone over the basics of how to enable a Linux system with Tomoyo. It is a very interesting addition to the Linux kernel and has some very nice features.

Potential Uses of Tomoyo

Highly secure systems - e.g. Android Phone
If you run through everything that runs on the Davlik VM you could theoretically validate everything that runs on it

Other potential uses, webservers, appliances, vpn, SCADA system, systems connected to dangerous networks, E.g. wireshark probes

VM hosts, could help prevent privilege escalation

Based on the idea that you can fully train system or create policies by hand

Some Tips

The interface for tomoyo-editpolicy quite simply sucks:
E.g. to get out of help you hit ? again, x exits the editor
Build up slowly with it. Locking down shells and so on
There was a graphical front end to the editpolicy called gpept. Did a quick search and haven't been able to find a real version of it available

Overhead

So what is the overhead of Tomoyo?

In terms of memory a small policy is around 100k

Quite simply the answer is that it depends. It depends on the complexity of your policy, what is running on the system and the number of rules being enforced

The assumed baseline is assumed to be negligible but depending on load and other conditions it can change

Summary

- Tomoyo is nowhere near as popular as SELinux
- SELinux is installed by default with Red Hat Linux
- Tomoyo is available for a lot of different distros
- Tomoyo is worthwhile and interesting in its own right
- The audit information that Tomoyo generates would be very interesting to see how it works in honeypots
- Tomoyo is Domain Type Enforcement not Role Based so it isn't as complete as SELinux in that respect

Q & A

Any Questions?

Thanks for listening.

Links

Tomoyo Homepage

http://tomoyo.osdn.jp

Arch Linux - Tomoyo Linux Page

https://wiki.archlinux.org/index.php/TOMOYO_Linux

Debian Linux - Tomoyo Linux Page

https://wiki.debian.org/Tomoyo

Links (Con'td)

Embedded Linux page for Tomoyo Linux

http://elinux.org/TomoyoLinux