BUILD A SECURITY SYSTEM WITH LIDS

Xie Huagang (xhg@ncic.ac.cn, http://www.lids.org) Tue Apr 18 00:02:12 CST 2000.

LIDS (Linux Intrusion Detection System) is a linux kernel patch to enhance the the linux kernel. In this article, we will talk about the LIDS including what it can do and how to use it to build a security linux system.

Contents

1 Why LIDS.

With increasing popularity of Linux on Internet , more and more security holes are found in the current GNU/Linux system. You may hear from the Internet that - There are bugs found in Linux, which will cause the system to be easily compromised by hacker.

Since the Linux is an art of open source community, security holes can be found easily and can also be patched quickly. But when the hole is disclose to the public, and the administrator is too lazy to patch the hole. It is very easy to break into the current system and it is worse that the hacker can get the root shell. With the current GNU/linux system, he can do whatever he want. Now, you may ask, what is the problem and what can we do?

What's wrong with the current GNU/Linux system.

- superuser (root) may abuse the rights

  Being a root, he can do whatever he want. Even the capability existing in the current the system. As a root, he can easily change the capability.

- Many system files can be changed easily.

  There are many important files, such as /bin/login, in the system. if the hacker come in, he can upload a changed login program to replace /bin/login , so he can re-login without any login name of password. But the files do not need to change frequently, unless you want to upgrade the system.

- Modules is easily used to intercept the kernel. Module is a good design for the linux kernel to make the linux kernel more modulized and more felixible. But after the modules inserted into the kernel, it will be part of the kernel and can do what the original kernel can do.

  Therefore some unfriendly code could be written as a modules and inserted into to kernel, the code can even redirect the system call and act like a virus.

- Process is unprotected.

  Certain processes, such as web server daemon, which are critical to to system is not under strict protection. Therefore, there are vulnerable to the attack of hackers.

With above description about insecurity thing, how can we build a security system? we must have a security kernel and then build our security system on top of it. This is what LIDS do.

2 Features about LIDS.

The Linux Intrusion Detection System is `a patch which enhances the kernel's security`. When it is in effect, chosen files access, every system/network administration operations, any capability use, raw device, mem and I/O access can be made impossible even for root. It uses and extends the system capabilities bounding set to control the whole system and adds some network and filesystem security features in kernel to enhance the security. You can finely tune the security protections online, hide sensitive processes, receive security alerts through the network, and more.

In short, LIDS provides `Protection, Detection and Response` to the intrusion in the linux kernel.

- Protection.

  LIDS can protect important files on your hard disk no matter what filesystem type they reside on, anybody include root can not change the files.

  LIDS can also protect the important process from being killed.

  LIDS can prevent RAW IO operation from an unauthority program. It can also protect your hard DISK,include MBR protection,etc.

- Detection.

  When someone scan your host, LIDS can detect it and inform the administrator. LIDS can also notice any activity on the system which violates the rules.

- Response.

  When someone violate the rules, LIDS can log the detail message about the violated action to the system log file which has been protected by LIDS. LIDS can also send the log message to your mailbox. In this case, LIDS can also shutdown the user's session at once.

3 Build a security linux system with LIDS

With the LIDS features, let's go and see how to build a security system with LIDS step by step.

3.1 Download LIDS patch and coresponsive official linux kernel

You can download lids patch from *LIDS Home* `<http://www.lids.org>` and *LIDS Ftp Home* `<ftp://ftp.lids.org>` and other mirror of LIDS around the world, check *LIDS Mirror* `<http://www.lids.org/mirrors/>` for the nearby mirror site.

The patch name will be lids-x.xx-y.y.y.tar.gz, x.xx represents the lids version and the y.y.y represents the linux kernel version.

You should download the coresponsive kernel version. For example, if you download the lids-0.9pre4-2.2.14.tar.gz, you should download the linux kernel 2.2.14 source code. You can download the kernel source from *Kernel FTP Site* `<ftp://ftp.linux.org>` or other mirror site of it.

```
1. uncompress the linux kernel source code tree.
# cd linux_install_path
# bzip2 -cd linux-2.2.14.tar.bz2 | tar -xvf -

2. uncompress the lids source code.
# cd lids_install_path
# tar -zxvf lids-0.9pre4-2.2.14.tar.gz
```

3.2 Patch LIDS to official linux kernel

After downloading the kernel source and lids, uncompress the source and lids. For example, if you download the lids-0.9pre4-2.2.14.tar.gz and linux-2.2.14.tar.bz2, then,

```
3. patch the lids to the linux kernel source code.

# cd linux_install_path
# patch -p0 </lids_install_path/lids-0.9pre4-2.2.14.patch
# cd linux
# make dep clean
# make bzImage
# make modules
# make modules_install

4. copy the bzImage to /boot/ and edit the /etc/lilo.conf

5 running /sbin/lilo to install the new kernel.

# /sbin/lilo
```

3.3 Configure the linux kernel

Now, it it time to configure the linux kernel, turn the following option on,

```
[*] Prompt for development and/or incomplete code/drivers
[*] Sysctl support
```

```
4. configurate the linux kernel
# cd linux
# make menuconfig or make xconfig
```

3.4 Initial the lids system

Now before reboot, you must configurate you lids system to meet you security need. You can define protected files, protected process,etc. In the next chapter, we will show your the details about this topic.

3.5 Reboot the system

After your configurate your system, reboot your system. When lilo appear, select the lids enable kernel to load. After then, you enter the wonderful world of LIDS.

3.6 Sealing the kernel.

After your system boots up, do not forget to seal the kernel with lidsadm, you can put the command in latest line of /etc/rc.local.

```
# /sbin/lidsadm -I -- -CAP_SYS_RAWIO -CAP_NET_ADMIN
```

You can check the LIDS HOWTO to view the details meaning of each options of lidsadm.

3.7 Online administration

After you seal the kernel, your system is now protected by LIDS. You can do some test on it. If you want to change some configuration, such as modify the capability option, you can change your lids security level online by providing a password.

```
# /sbin/lidsadm -S -- -LIDS
```

4 Configurate LIDS

In this chapter, we will show you how to configurate the lids.

4.1 Protect you files.

Firstly, you must determine which files you will protect. In most case, you may protect the system binary files and system configuration files, such as /usr/, /sbin/, /etc/, /var/log/.

Secondly, you must decide the way to protect the files. LIDS provide 3 protection type:

- Read Only Files. The files marked with Read Only means that nobody can change the files. We can think that the following files are in this catalog, /etc/passwd, /bin/passwd, etc.

  ```
  USAGE:
  lidsadm -A -r filename_to_protect

  example,

  1. to protect the whole /sbin/ as read-only.

   # /sbin/lidsadm -A -r /sbin/

  2. to protect /etc/passwd as read-only

   # /sbin/lidsadm -A -r /etc/passwd
  ```

- Append Only Files.

  Most of the append only files are system log files, such as /var/log/message ,/var/log/secure. The files can only open with append mode and can not truncate or modify its previous contents.

  ```
  USAGE:
  lidsadm -A -a filename_to_protect

  example,

  1. to protect the system log files

  # /sbin/lidsadm -A -a /var/log/message
  # /sbin/lidsadm -A -a /var/log/secure
  ```

```
2. to protect the apache httpd log files

# /sbin/lidsadm -A -a /etc/httpd/logs/
# /sbin/lidsadm -A -a /var/log/httpd/
```

Here is the example from LIDS HOWTO by Philippe Biond,

```
lidsadm -Z
lidsadm -A -r /boot
lidsadm -A -r /vmlinuz
lidsadm -A -r /lib
lidsadm -A -r /root
lidsadm -A -r /etc
lidsadm -A -r /sbin
lidsadm -A -r /usr/sbin
lidsadm -A -r /bin
lidsadm -A -r /usr/bin
lidsadm -A -r /usr/lib
lidsadm -A -a /var/log
```

4.2 Protect your process.

LIDS can protect the process whose parent is init(pid=1), you must seal the kernel with a specified option as below.

```
# lidsadm -I -- +INIT_CHILDREN_LOCK
```

4.3 Protect with capability.

Capabilities are like privileges you can give a process. A root process has all the capabilities. But there exists a capabilities bounding set. In a normal kernel, when you remove a capability from the bounding set, nobody can ever use it again, until next reboot. (see http://www.netcom.com/ spoon/lcap for the normal use).

LIDS modifies this behavior to enable you to switch theses on and off, whenever you want. An access to the `/proc/sys/kernel/cap_bset` is trapped and raise a security alert. lidsadm performs all the job.

You can list all capability in LIDS by running lidsadm, and you can see what the exactly meaning of each capability.

We here discuss two of them,

4.3.1 CAP_SYS_RAWIO

With this capability on, we can allow ioperm/iopl and /dev/port access, allow /dev/mem and /dev/kmem acess and allow raw block devices (/dev/[sh]d??) acess

When we disable this capability, we can make all process on the system has no any right to the raw device, such as runing lilo.

But some process may want this capability to run, such as XF86_SVGA. In this case, we can let the program in the exception list when compile the kernel.

4.3.2 CAP_NET_ADMIN

This capability get the following ability,

```
interface configuration
administration of IP firewall, masquerading and accounting
setting debug option on sockets
modification of routing tables
setting arbitrary process / process group ownership on sockets
binding to any address for transparent proxying
setting TOS (type of service)
setting promiscuous mode
clearing driver statistics
multicasting
read/write of device-specific registers
```

For the security reason, we should disable this to disallow changing network configuration. When it disallow, the firewall rules will not allow to change.

4.3.3 Choose the capability and sealing the kernel

You should choose what capability you should disallow when sealing the kernel. Here we give an example.

You may put it in a rc script (rc.local, /etc/init.d/lids, /etc/rc.d/init.d/lids, etc.) depending upon your distribution and the way you administrate your system. The command is, for example :

```
lidsadm -I -- -CAP_SYS_MODULE -CAP_SYS_RAWIO -CAP_SYS_ADMIN \
              -CAP_SYS_PTRACE -CAP_NET_ADMIN \
              +LOCK_INIT_CHILDREN
```

4.4 Network Security.

LIDS provide some network security enhancement.

4.4.1 network security with capability

With the capability, we can enhance the network security. such as anti snifferring, can not bind to the port lower than 1024, can not change the firewall and routing rules.

so, what I suggest is to view each capability meaning carefully.

4.4.2 Scanner detector in kernel

LIDS provide a scanner detector in kernel in order to detect who had scan your system. The scanner can detect half-open scan, normal scan etc. using tools like nmap,satan can be detected by the detector.

It is useful when raw socket is disable. In this case, some user space detector based on snifferring will not work. And the detector does not use any socket, it will be more secure than a user space detector.

If you want this feature, you should select it on when compile the kernel.

4.5 Intrusion Responsive system.

When LIDS detect some thing violate the rules defined, it can response to the action by following method.

4.5.1 Logging the message

When someone violate rules, lids_security_log will log a message the klogd, the logging also have the ability to anti_logging_flood. You can set it when compile the kernel.

4.5.2 Logging the message via mail server

Now, LIDS has a new feature to mail the message to your mail account. you can define the mail server IP, the out-coming mail address,etc, when compile the kernel.

4.5.3 Shutdown the console

When user violate the rules, the console will shutdown the user's console.

5 Thanks.

First of all, I want thank my friend, Kate lee , who always encourage me to write document of that, this document is dedicated to her.

I also want to thank Philippe Biond and Christophe Long who give many help to the project. Without them, the project can not develop so well.

Many thanks must go to all the LIDS users, without their contribution and discussion, LIDS can not has so many great ideas.