ipv6mon: An IPv6 Address Monitoring Tool

Introduction

IPv6 hosts typically configure their IPv6 addresses by means of a mechanism known as “StateLess Address Auto-Configuration” (SLAAC), which results in de-centralized address assignment (i.e., each node assigns itself its own addresses).

> In SLAAC, local routers advertise the IPv6 prefixes that should be used for configuring IPv6 addresses, and hosts configure their addresses by selecting an Interface-ID, based on local policy, for each of those prefixes.

A common problem when deploying IPv6 networks is correlating the activities of local nodes, since there is no single system that keeps a record of which addresses have been used by which system at which moment in time. For example, if a local node gets infected with malware and tries to infect other nodes on local or remote networks, it may be impossible (given the node's IPv6 address), to identify which physical system is the one infected with malware.

ipv6mon aims at solving the aforementioned problem, by maintaining an historical record of which IPv6 addresses have been in use by each node in the local network.

One of the design goals of ipv6mon is that it should be a lightweight application with minimum processing requirements, and that it should be able to detect all/most addresses "in use" at the local network, even if they are not actively used for sending packets. To achieve this goal, ipv6mon listens to a specific subset of local traffic (rather than listening to all local packets), and also employs active-probing to identify active addresses (even if they are not actively used for sending packets) and to detect when such addresses are no longer used.

ipv6mon may be run on any system that connects to the local network (i.e., it need not run on a "mirror port" of a local switch). It has been tested to compile and operate correctly in Ubuntu, Debian GNU/Linux, and FreeBSD and NetBSD.

Installation notes

An installation script “INSTALL-Debian-Ubuntu.sh” is provided for easy installation in Ubuntu and Debian GNU/Linux systems. An installation script “INSTALL-FreeBSD.sh” is provided for easy installation in FreeBSD systems. Additionally, the scripts “UNINSTALL-Debian-Ubuntu.sh” and “UNINSTALL-FreeBSD.sh” are provided such that ipv6mon can be easily uninstalled off the aforementioned systems.

For other systems, installing ipv6mon will typically involve the following steps:

1) Compiling ipv6mon
2) Creating a special user (ipv6mon) and group (ipv6mon) for the ipv6mon tool
3) Setting the appropriate owner and permission bits to the executable file (ipv6mon)
4) Copying the ipv6mon executable file (ipv6mon) to an appropriate directory
5) Editing the sample ipv6mon configuration file (ipv6mon.conf)
6) Setting the appropriate permission bits and owner to the configuration file (ipv6mon.conf)
7) Copying the ipv6mon configuration file (ipv6mon.conf) to an appropriate directory

8) Updating the system configuration such that ipv6mon is started when the system is bootstrapped
9) Configuring your system such that the log files are automatically rotated

The aforementioned steps are described in more detail in the following subsections.

1) Compiling ipv6mon

ipv6mon can be compiled using the gcc compiler, with the following command:

```
gcc ipv6mon.c -Wall -lpcap -o ipv6mon
```

Note: The libpcap library should be previously installed on your system.

2) Creating a special user and group for ipv6mon

While ipv6mon needs superuser privileges to bootstrap, it releases such privileges once they are no longer needed. ipv6mon will switch to the unprivileged user and group specified by the UnprivilegedUser and UnprivilegedGroup variables, respectively, in the configuration file.

Therefore, during the installation process, the “ipv6mon” user and group should be created on the local system. The “ipv6mon” user should have no home directory, should not be allowed to login, and should not be assigned a real login shell.

The aforementioned user can usually be created with the useradd(8) or adduser(8) commands. Please consult the your system's manual pages, since the syntax of these commands vary from one system to another.

3) Setting the appropriate permission bits and owner to the executable file (ipv6mon)

The ipv6mon executable file should be “readable”, “writeable”, and “executable” by the owner, and should be only “readable” and “executable” by the group and others. This can be achieved by means of the following command:

```
chmod 0755 ipv6mon
```

The owner and group of the executable file should be “root” and “root”, respectively. This can be achieved with the following command:

```
chown root:root ipv6mon
```

4) Copying the ipv6mon executable file (ipv6mon) to an appropriate directory

You may want to move the executable file "ipv6mon" to an appropriate system directory. For example, to “/usr/sbin”. This can be achieved with the following command:

```
cp ./ipv6mon /usr/sbin/
```

5) Editing the sample ipv6mon configuration file (ipv6mon.conf)

At the very least, you should edit the sample configuration file (ipv6mon.conf) such that an appropriate network interface card is selected for IPv6 address monitoring.

6) Setting the appropriate permission bits and owner to the configuration filele file (ipv6mon.conf)

The configuration file should be owned by the user and group “root”. This can be achieved with the following command:

```
chown root:root ipv6mon.conf
```

Additionally, the configuration file should be readable by all, but writable only by its owner. This can be achieved with the following command:

```
chmod 0644 ipv6mon.conf
```

7) Copying the ipv6mon configuration file (ipv6mon.conf) to an appropriate directory

While it is possible to specify a pathname for the configuration file at runtime (with the “-c” option), it is generally desirable to make the edited configuration file available on the default pathname (/etc/ipv6mon.conf).

This can be achieved with the following command:

```
cp ./ipv6mon.conf /etc/
```

8) Updating the system configuration such that ipv6mon is started when the system is bootstrapped

Your system configuration should be updated such that ipv6mon is automatically started when the system is bootstrapped. This may involve adding corresponding entries in the in the “/etc/rc.d/” directory, editing the “/etc/rc.local” file, or the like. Please consult your system manuals about the necessary steps to be performed.

9) Configuring your system such that the log files are automatically rotated

Since the corresponding log file (typically “/var/log/ipv6mon.log”) will grow over time, you may want to configure your system such that the log file is rotated. This will usually involve configuring newsyslog(8) or logrotate(8).

Log-rotation tools typically archive the existing log file, and require the corresponding daemon to re-create and/or re-open its log file. However, since ipv6mon releases superuser privileges once a few initial actions are performed, it cannot perform such operations. A workaround for this is to have the log-rotation tool “truncate” the original log file, such that ipv6mon does not need to re-open or re-create it. For example, this can be achieved in Debian and Ubuntu systems with the “copytruncate” option of logrotate(8) tool.

Credits

The ipv6mon tool and related manuals were produced by Fernando Gont <fgont@si6networks.com>.

License