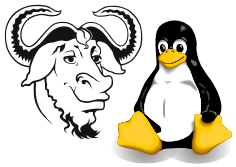

SNMP VACM and USM Tutorial

1 Aims

The main aims of the activities today are:

- To understand how to configure and install an SNMP agent on your computer;
- To understand how to use VACM to provide access control to any chosen set of variables;
- To understand how to create USM accounts and use them with strong authentication and encryption.

2 Procedure

The agent is called `snmpd`. The configuration for the agent is in the file `/etc/snmp/snmpd.conf`. Your root password is as described in `http://nicku.org/snm/lab/regular-expressions/regular-expresssions.pdf`: `)3SnhGxv9`. Set up `sudo` as described in `http://nicku.org/ossi/lab/sudo/sudo.pdf`. Edit your login script:

```
$ emacs ~/.bash_profile &
```

and add the following two lines:

```
export http_proxy=http://dproxy.vtc.edu.hk:8080/
PATH=$PATH:/sbin:/usr/sbin
```

and save (Control-x Control-s). Then exit (Control-x Control-c). Finally *source* your login script with

```
$ . ~/.bash_profile
```

1. Install the updated software packages:

   ```
   $ sudo yum install net-snmp\*
   ```

   Actually, I think that it is probably best to apply *all* the updates:

   ```
   $ sudo yum -y update
   ```

2. Edit the configuration for the SNMP agent:

   ```
   $ xhost +localhost
   $ sudo -v
   $ sudo emacs /etc/snmp/snmpd.conf &
   ```

The first line allows users other than your own account (such as the user root) to display graphical objects on your local X server.

The second line starts a new five-minute password free period for `sudo`. If the five-minute period has expired, then the editor cannot start in the background; `sudo` will wait for you to bring it to the foreground by typing `fg`, so that you can enter your password. Typing `sudo -v` simply avoids this inconvenience.

3. Start the agent:

```
$ sudo /sbin/service snmpd start
```

4. Enable the agent to always start when the computer boots:

```
$ sudo /sbin/chkconfig snmpd on
$ /sbin/chkconfig snmpd --list
```

The first command ensures that the next time a computer boots on your hard disk, it will start the agent `snmpd` whenever it moves into runlevels 3, 4 or 5.

The second command lists which runlevels `snmpd` will start at, to confirm to yourself that the previous command worked.

5. Note that any time you change the configuration for the agent (by editing `/etc/snmp/snmpd.conf`), you will need to restart the agent to get it to read the new configuration with:

```
$ sudo /sbin/service snmpd restart
```

2.1 How do I Tell If It Worked?

When you restart the agent, any *syntax errors* will be shown in the system logs. You should open another window and watch the logs with the command:

```
$ sudo tail -f /var/log/messages
```

How do you test your VACM Configuration? To test *read-only access control*, use `snmpget`, `snmpgetnext` or `snmpbulkwalk` to access the variables using the security name that belongs to the group you are controlling access to.

To test *read-write access control*, use `snmpset`.

See `man snmpcmd` for details about how to specify USM usernames and keys for authentication and "privacy" (encryption). See also slide §33 in the SNMPv3 lecture notes for examples of using `snmpget` on USM accounts.

3 Questions

Refer to my notes about VACM and USM.

1. Create two security names with different community strings using `com2sec` in your agent's configuration, in `/etc/snmp/snmpd.conf`.

2. Create a view that includes all the 22 columns for the first network interface in the `ifTable` (i.e., from IF-MIB::ifEntry.1 to IF-MIB::ifSpecific.1), and another view that includes all the 22 columns of the second network interface in the `ifTable` (i.e., from IF-MIB::ifEntry.2 to IF-MIB::ifSpecific.2).

3. Create two groups that map the security names to each of the two access control entries in your agent's configuration, in `/etc/snmp/snmpd.conf`.

4. Implement read only access for each of your two users to each of the two views you created earlier for question 2.

5. Implement this and verify that the access control works. Write here how you verified that it works.

6. Now create two USM users, and configure them so that access works to the two views you created. Comment out the `com2sec` mappings, so that only USM access is permitted. Demonstrate that this works.