

Net-SNMP Notes

1 General

Net-SNMP can be downloaded from <http://net-snmp.sourceforge.net>. This package was originally called ucd-snmp (for UC Davis), so you'll often see it referred to as ucd-snmp instead of net-snmp. This package works on both Linux (Slodev1) and Windows XP (Pismo), although it primarily supports Linux/*NIX systems.

ucd-snmp refers to version 4.x and earlier, while version 5.0 and forward are called net-snmp. The major difference between version 4 and 5 is code cleanup and added functionality added to the agent.

There are a number of utilities that come with net-snmp, but only snmptranslate, snmpget, and snmpset are discussed here.

See the Net-SNMP website (<http://net-snmp.sourceforge.net/tutorial-5>) for several good tutorials on using the SNMP tools.

NOTE: ucd-snmp is distributed with the Intersil AP DK. In fact, it looks like the SNMP agent that runs on the Intersil AP is the ucd-snmp agent.

2 Installing net-snmp

Even though net-snmp is targeted for *NIX platforms it is possible to run it ran from a Windows host. The Linux install is pretty simple, but the Windows install is more involved. The install and environment setup process for both platforms is described below.

2.1 Linux Install

The Linux install process is straightforward. If you need to install on Linux, then just follow the install directions provided by net-snmp. When finished with the install skip down to 2.3 Environment Set Up

2.2 Windows Install

Setup was OK on Windows XP, but not as smooth as Linux. Before installing and building net-snmp, the Windows Simple Network Management Protocol package was installed. It's not clear that it is necessary to install this package before installing net-snmp, but it was done just to be safe.

The Windows SNMP package can be installed by opening "Add or Remove Programs" from the Control Panel. From the "Add or Remove Programs" dialog box click on

"Add/Remove Windows Components" and then double click (but do not check) "Management and Monitoring Tools". From the "Management and Monitoring Tools" dialog box check "Simple Network Management Protocol" and then press OK. Press Next, OK, or whatever buttons are necessary to complete the install. You will need the Windows XP install disk to complete this process.

net-snmp can be built with a Microsoft Visual C++ project or from Cygwin. If you build from Cygwin then you'll need the GNU compiler and potentially other development packages. Since I've had mixed luck with some of the Cygwin development tools I opted to build net-snmp from the MSVC++ project.

I ran a batch build, which generated the debug and release net-snmp tools and libraries. The dependencies are not set up quite right for this project so several tools failed to build the first time, but I could build all of them the second time I ran the batch build.

The file net-snmp-config.h contains a few constants that you may want to edit. First the constant DEFAULT_MIBDIRS expects MIBs to be located at /USR/MIBS. This was left as the default option on Pismo, so if you need to add a new MIB then it should be added to this directory (see 2.3 Environment Set Up for more information).

Once net-snmp successfully built on Pismo, the release tools were copied to the Cygwin /usr/bin directory (C:\cygwin\usr\local\bin).

2.3 Environment Set Up

For both the Linux and Windows hosts the MIB for the 802.11 devices you want to access needs to be made available for the net-snmp tools. The examples below assumes you are using the MIB for the Linksys WAP11. The Linksys WAP11 MIB is in the Linksys WAP11 directory on Pismo at *C:\Program Files\Linksys WAP11\WAP11 SNMP\AT76C510.mib*. This is actually an ATMEL MIB since Atmel is the OEM of the chipset used in the Linksys WAP11.

First add the Atmel MIB (AT76C510.mib) to the net-snmp MIB directory. On slodev1 (and Linux in general) there are two options: */usr/local/share/snmp/mibs* or *\$HOME/.snmp/mibs*. Use the first option if you want all users on slodev1 to have access to the MIB, otherwise use the second option if only you will use the MIB.

On Pismo put the MIB in the directory *C:/usr/mibs*. Unless you change the DEFAULT_MIBDIRS constant in the file net-snmp-config.h then MIBS will always be stored in the *C:/user/mibs* directory on a Windows machine.

2.3.1 MIB Clean Up

Net-SNMP is a bit fussy about the MIB format so edit the MIB file to remove any comment lines with more than 3 '-' (dash) characters.

For example:

```
-----Authorized Mac Addresses-----
```

was replaced with

```
-- *****Authorized Mac Addresses*****
```

Also net-snmp does not like empty variable blocks so the last 2 traps in the Atmel MIB file were commented out.

These changes have already been made to the Atmel MIBS on slodev1 and Pismo. But just keep in mind that you may need to make similar changes to any new MIBS that you add.

2.4 Loading the MIB

By default, net-snmp will not load the Atmel MIB. There are several options for instructing net-snmp to load a MIB, but the easiest method is to set the MIBS environment variable. For both Linux and windows systems set MIBS to "+ATMEL-MIB". Using the bash environment this looks like:

```
export MIBS="+ATMEL-MIB
```

Note that the MIB name is not the name of the file. net-snmp assumes all files in its MIB directory are MIB files. The name of the MIB is specified by the first line in the MIB that ends with the string "DEFINITIONS ::= BEGIN". For the Atmel MIB the name specified by this line is "ATMEL-MIB", which is the name of the MIB we've used to set the MIBS environment variable.

NOTE: Don't keep multiple copies of the same MIB file in the MIBS directory since net-snmp doesn't have enough sense to use the file "AT76C510.mib" instead of "AT76C510.mib.backup"! In general, if you have two copies of the MIB file in the same directory net-snmp will use the one you don't want it to use. Plus the net-snmp error messages are not very descriptive so it may take awhile to notice that an error is caused by something as simple as net-snmp reading the wrong MIB file.

3 net-snmp Tools

The following tools are included in the net-snmp package:

- snmptranslate
- snmpget
- snmpgetnext
- snmpwalk
- snmptable
- snmpset

- snmptrap

Only the snmptranslate, snmpget, and snmpset tools have been successfully used. In general, when working with the Linksys WAP11 remember that the AP's address is 128.18.233.50 and that its community identification string is "orchid". Also, all SNMP commands are in SNMP version 1 format. The sections below describe how to use these values to get and set SNMP information from the Linksys WAP11.

You may also want to see my SNMP notes for general information about SNMP.

3.1 SNMPTRANSLATE

snmptranslate is used to browse a local MIB. It's a good tool for checking that the desired MIB is loaded and can be found by net-snmp. Plus its good for making sure that you understand the structure of the target MIB.

3.2 SNMPGET

Snmpget retrieves values from the Linksys WAP11 agent. To get the value of the WAP11's IP address then you would request the value of the operIPAddress object like this:

```
snmpget -v 1 -c orchid 128.18.233.50 operIPAddress.0
```

The -v argument is used since this is a SNMP version 1 command. The community flag (-c) is specified with a value of "orchid", and the agent location is 128.18.233.50, which is the same value that should be returned from the AP.

To obtain other values from the AP, you can substitute operIPAddress.0 with other objects defined in the Atmel MIB. Other possibilities include operAccessPointName (name of the access point), sysDescr (a description of the AP including the vendor name and firmware number), or operSSIDBroadcasting (indicates if the WAP11 is broadcasting the SSID or not)

3.3 SNMPSET

snmpset is used to set object values on the Linksys WAP11. snmpset can be used to change the name of the WAP11 or to toggle the broadcasting of the SSID. To stop the broadcasting of the SSID use snmpset as follows:

```
snmpset -v 1 -c orchid 128.18.233.50 operSSIDBroadcasting.0 i 2
```

This command is similar to snmpget since the same values for the SNMP version (-v 1), community (-c orchid) and agent (128.18.233.50) are used. The difference is in the last two options, which include the data type and the new value.

As defined in the ATMEL-MIB the `operSSIDBroadcasting` object is an integer data type. When this object is set to 1, then the SSID is broadcasted. If this object is set to 2 then the SSID is not broadcast.

Data types used by other objects include:

- i – integer
- u – unsigned integer
- t – timeticks
- a – IP Address
- o – object ID
- s – string
- x – hex string
- d – decimal string
- U – unsigned int64
- I – signed int64
- F – float
- D – double

When setting a string data type enclose the new value in quotes.

3.4 Other SNMP Tools

The tools `snmpgetnext`, `snmpwalk`, and `snmptable` do not work with the Linksys WAP11. They could be useful tools, but they are not necessary to test the SNMP capabilities of the Linksys WAP11 or other SNMP enabled Access Points, so not much time was spent getting these tools to work.

Another `net-snmp`, `snmptrap`, has not been tested due to the complexity of dealing with traps.

4 Sample Commands

```
snmptranslate -m+ISL-GLOBAL-REG -Td 1.3.6.1.4.1.937.2.1.2.2
snmptranslate -m+ISL-GLOBAL-REG 1.3.6.1.4.1.937.2.1.2.2
snmptranslate -m+ISL-GLOBAL-REG -Of 1.3.6.1.4.1.937.2.1.2.
snmptranslate -m+ISL-GLOBAL-REG -Of 1.3.6.1.4.1.937.2.1.2.
snmptranslate -On ISL-GLOBAL-REG::islGeneric.1.2.6.0
snmptranslate -On IP-MIB::ipAdEntAddr
```

```
snmpget -v 1 -c password -m+ISL-GLOBAL-REG 224.0.1.43 1.3.6.1.4.1.937.2.1.2.2.0
snmpwalk -v 1 -c public -m+ISL-PRISM-MIB 224.0.1.43 1.3.6 > isl-prism.txt
snmpgetnext -v 1 -c public -m+IEEE802dot11-MIB -Cf 224.0.1.43 1.2.840.10036.1.1.1.1
```