Chapter 5

Installing Snort and MySQL for Windows

In This Chapter

- Getting to know Snort for Windows
- Setting up Snort for Windows 2000
- Setting up MySQL for Windows 2000 and Snort

For an average Windows user, installing Snort is a little more of a headache than for your average Linux user. This is because Snort was developed initially for open-source Unix-like platforms such as Linux, and if you are at all familiar with Linux, you know what that means: command-line options and text-based configuration files. For a Windows user who's used to point-and-click configuration, command-line is a little intimidating. Add to that the fact that there's little supporting documentation for the Windows platform on Snort's Web site or the rest of the Internet, and you have all the makings of a bumpy ride.

Never fear: This chapter gives you step-by-step installation instructions for getting your Snort IDS up and running on Windows.

The Windows Snort IDS Box

These are the minimum requirements for a Windows Snort box:

- A PC running Windows NT 4.0, Windows 95, Windows 98, Windows 2000 (Server or Professional), Windows XP (Home or Professional), Windows 2003 Server
- A packet-capture driver for Windows (WinPcap is really your only choice)
- One or more network interface cards (NICs) and a network connection
- Snort

The preceding requirements are definitely the *minimum* requirements for running Snort on a Windows box: You can get Snort up and running with that configuration. You can also drive a front-wheel-drive car with just the two front wheels, but you're not going to get very far, your tail-end will spew a lot of sparks, and you might explode along the way. The point is that the minimum requirements are not necessarily the best configuration. In the following sections we go over specific recommendations for the Windows OS, logging database, and system resources.

Choosing your Windows OS

Just because Snort can run on practically any 32-bit version of Windows, doesn't mean you should run Snort on just any version of Windows. We recommend running Snort on either Windows 2000 Professional or Windows XP Professional for the following reasons:

- Windows 2000 and XP Professional are more secure and stable than the "home user" Windows systems, such as Windows 98, Windows ME, or Windows XP Home Edition. This is due to features such as the NTFS filesystem, better multitasking, and better memory management in 2000 and XP Professional.
- The "home user" Windows systems, such as Windows 98, Windows ME, or Windows XP Home Edition are not suitable for running a Web server such as Internet Information Services (IIS). A Web server is required for the ACID visualization console we cover in Chapter 7.
- The "home user" editions of Windows only support a single processor, whereas Windows 2000 and XP Professional support dual processors.
- Windows 2000 and XP Professional are still supported by Microsoft, unlike Windows NT 4.0 (or earlier versions of NT).
- Windows 2000 and XP Professional are cheaper alternatives than Windows 2000 Server or Windows 2003 Server.

In some high-performance environments the server-class versions of Windows 2000 and 2003 might make more sense, such as when you want to take advantage of systems that have more than two CPUs.

The minimum configuration only gets you text-based logging and alerts, which can be hard to manage. In the long run, we want to be able to classify alerts and use reporting and visualization tools such as the ACID console we cover in Chapter 7. In order to do this, we need to run an RDBMS (Relational Database Management System, a fancy name for a database program).

MySQL, your SQL

The RDBMS we chose is MySQL. MySQL is a free database that works on a number of platforms, including Windows. As a Windows user you might already be familiar with some of the Microsoft database products, such as MS SQL and Access, and are wondering why we aren't using those. MySQL has a number of things going for it as a backend database for Snort:

- Snort can log directly to MySQL natively, as the alerts come in. Snort can't currently log in real-time to Access databases.
- Snort's unified logging output can be converted directly to MySQL using the Barnyard utility (covered in Chapter 14). Barnyard cannot currently convert Snort's unified logging output directly to Access formats.
- MySQL is supported by many extra Snort tools, including the ACID visualization console we cover in Chapter 7. ACID currently does not support Access databases.
- Did we mention that MySQL is free? MS SQL and Access licenses aren't free, which can increase the cost of your Snort IDS if you don't already own those licenses.

If you've never used MySQL or any other RDMS before, don't worry. You don't need to be a database guru or even understand SQL queries to get Snort up and running with MySQL. We provide instructions to get Snort logging to MySQL under Windows.

Two resource hogs: Windows and Snort

All Windows-based operating systems have high base hardware requirements relative to other operating systems, even with as much unnecessary stuff removed as possible. When it comes to recommended hardware, for Snort, the faster and more the better. Snort needs as much processor speed and memory you can throw at it, relative to the activity on your network:

- If Snort runs out of resources, it drops packets; it won't analyze all of the network packets that come under its nose. With Snort dropping packets, the entire purpose of an IDS is defeated; an attack on your network or hosts can come at any time. (Murphy's Law says the attack will probably come when your IDS is overloaded.)
- If you plan to run MySQL (or another database system), IIS (or another Web server), and ACID (and all its dependencies) on the same computer as Snort, consider fielding a *very* fast system.

For high-traffic production networks, you'll get the best performance from Snort by running the database, Web server, and sensor on different computers. Look at your network traffic and the requirements for the OS you select before setting up a Snort system. Chapter 3 should give you a better idea about how to size your Snort system to your particular environment.

Program storage requirements

With MySQL and support programs, the full Snort complement could fill as much as 60MB of hard drive space. That's not a huge amount of space by today's I-need-hundreds-of-gigs-just-for-my-downloaded-music standards, but that figure is only for the software itself, not the data you're going to collect using it. The Snort executable takes a measly 400KB of disk space. The entire Snort package takes 5.8MB on initial install.

Data storage

Your data storage requirements depend on what you do with the data:

- If you're capturing all packets on your network and storing them with Snort (not something you'd normally keep around forever, though) your storage needs will grow exponentially, daily.
- If you are running a single sensor and looking for only a few alerts or using a small rule base, you don't need much disk storage space.

In our testing environment, we captured alerts off of the basic Snort rules, and these alerts average about 5KB per alert in the text alert format. Though the size of the alert may be pretty standard, how many are generated on your network and how many are captured are up to you. Chapter 8 gives more detailed guidelines on rules and how to use them to maximize your Snort system.

Partition configuration

When installing your Windows operating system, set up at least two partitions on the hard drive:

- A small partition sized for the OS and applications running on your computer. By "small" we mean large enough to hold the Windows operating system, which can take as much as 3GB of disk space. We recommend making this partition at least 6GB in size.
- A larger partition for data depending on the amount of data you plan for Snort to capture. This is where your Snort logs and alerts go, so the amount of space varies depending on your network. It's a good idea to make it as large as you can.

Separate OS and data partitions keep the partitions from corrupting each other in case one fills up, and makes it much easier to back up to the partitions individually on separate schedules.

For extra security on your Web server we recommend having your IIS document root on its own partition, too.

Keeping Your Windows Locked

Before installing Snort and any other components, it's important to lock down your Windows system. After all, what good is a Snort IDS that's been compromised by an attacker? No good at all.

Hardening any Windows OS has become more difficult over the past few years, as more and more applications are integrated with the base operating system. Even so, following the guidelines and recommendations set forth in this section will help you secure your Windows-based Snort system.

Limit physical access

Physically secure the system in the following ways:

- Locate your Snort sensor in a secure area, accessible only to people who need physical access to the machine.
- Configure the system to boot only from the hard drive. You don't want someone bypassing Windows' security controls simply by booting off a floppy disk or CD-ROM, or even a keychain-sized USB drive!
- Consider using a system with a locking front panel that prevents an unauthorized person from booting from a floppy disk or CD-ROM.

Nobody should have access to the console of the Snort IDS sensor but you!

Tighten OS access control

Limiting the users who can log on to your system and having a good password policy are imperative. Here are a few suggestions for keeping your accounts secure:

- Set up a strong password policy on the system.
  - Always use a complex password that uses a combination of upper and lower case letters, numbers, and special characters (*!#$).
  - Use passwords of eight characters or more.
  - Enable logging of login attempts, failures, and successes.
- You need one user on this system: the Administrator.
  - Immediately change the Administrator account name.
  - Rename and disable the Guest account (you can't remove it).
  - Remove all other accounts.

Nothing makes a hacker's job easier than choosing a simple word or name for your password, or allowing guest access to your system. So, don't make a hacker's day: Follow the preceding account lockdown suggestions.

Harden the OS

Hardening an OS means to take measures to increase security and reduce vulnerabilities that go beyond the default installation of the OS. Since Windows is a general-purpose OS designed for user-friendliness, there are many features turned on by default that aren't required on a Snort system. Here are a few suggestions for hardening a Windows Snort IDS box:

- Install only components that are absolutely necessary to run the OS.

  Windows operating systems install many programs that you don't need for a Snort IDS. Most notable are such applications as Windows Media Player and Outlook Express. Install *nothing* extra and add what programs you need, later.

  When given the option, *just say no.*
- After installing Windows, turn off all unneeded services.

  Windows runs a plethora of services in the background that aren't needed for every implementation of the OS. Figure out what you need and turn off the rest.
- Disable unneeded network protocols. All you need is TCP/IP. That's it. Everything else*: out the window!*

Use `netstat` from the command line on your Windows box to list the network services that are listening (or connected) at any given time. To use `netstat` to list all the listening ports by protocol, open a command window and type

Patch, patch, and patch again

We can't emphasize enough the importance of patching on Internet-facing Windows systems! All the infamous destructive Windows worms of 2003 — Slammer, Blaster, Nachi — used known holes for which patches already existed.

If you're concerned about patching a production Snort IDS, set up a second Windows Snort box as a test bed. Set up that box to automatically use Windows Update to detect and download the latest critical patches, and test it first. If everything continues to work perfectly after patching, do it on the production system. Same goes if you're in an enterprise environment and use Microsoft's Systems Management Server (SMS) or Software Update Services (SUS) for patching: Test, then deploy.

Just do it, and plan to do it regularly (Microsoft is currently releasing patches once a month, so this makes it much easier to plan). The comfort zone between the discovery of a vulnerability and the release of a worm is rapidly shrinking.

```
netstat -an
```

- Conduct all remote communications to and from the sensor with secure protocols and applications, such as IPSec, SSL, and `ssh`.
- Apply all security updates, patches, and service packs.

  Maintenance is imperative. Regularly check for new security updates, patches and service packs. New Windows-specific exploits hit the wire all too frequently.

There are reams of information available on the Internet for securing Windows systems. Here are a few of our favorite Windows security resources:

- The security wizards at SANS list the Top 20 critical security vulnerabilities for Windows (at `http://www.sans.org/`).
- The Center for Internet Security (a group that includes SANS, government agencies, and private industry) has a security benchmarking tool at `http://www.cisecurity.org/`.
- Microsoft's Baseline Security Analyzer and IIS Lockdown Tool is available at its Web site, `http://www.microsoft.com/`. Always get the latest versions.

Hardening your Windows Snort IDS is an ongoing process.

Installing the Base Snort System

Installing the base Snort system requires two components: the WinPcap packet capture library, and the Snort IDS program itself. In the following sections we configure and install both WinPcap and Snort.

WinPcap

WinPcap (Windows Packet Capture Library) is a *packet-capture driver*. Functionally, this means that WinPcap grabs packets from the network wire and pitches them to Snort.

WinPcap is a Windows version of `libpcap`, which is used for running Snort with Linux. For more on `libpcap`, see Chapter 4.

Functions

The WinPcap driver performs these functions for Snort:

- Obtains a list of operational network adapters and retrieves information about the adapters.
- Sniffs packets using one of the adapters that you select.
- Saves packets to the hard drive (or more importantly for us, pitches them to Snort).

Installation

The installation and configuration of WinPcap is dead easy, with almost no intervention by you:

1. **Download the latest installation file from**

   ```
   http://winpcap.polito.it/install/default.htm
   ```

   The installation file is generally called something like WinPcap_3_0.exe.

2. **Double-click the executable installation file and follow the prompts.**

   WinPcap installs itself where it belongs.

Snort calls WinPcap directly on any of the functions to grab and analyze network packets. If the driver did not install properly, Snort does not function.

Accept no substitutes for Windows

These tools verify that the programs you download from the Internet haven't been tampered with by a miscreant (this process is called "integrity checking"). We highly recommend that you use them.

- A Windows equivalent for `md5sum` is MD5summer. MD5Summer is free and has an easy-to-use GUI interface for generating MD5 checksums. It can be found at `http://www.md5summer.org/`.
- A Windows binary for GnuPG can be found at `http://www.gnupg.org/(en)/download/index.html`.

Time for a Snort

Snort.org distributes a convenient install package for Windows available at its Web site:

`http://www.snort.org/dl/binaries/win32/`

Download this package (generally called `snort-2_1_0.exe`) and perform the following steps to install Snort:

1. **Double-click the executable installation file.**

   The GNU Public License appears.

2. **Click the I Agree button.**

   Installation Options window appears.

3. **In the Installation Options dialog box, click the appropriate boxes to select from among these options:**

   - **I do not plan to log to a database, or I am planning to log to one of the databases listed above.** Choose this option if you are not using a database or if you are using MySQL or ODBC databases. Snort has built-in support for these databases, and for our example, we chose this option.
   - **I need support for logging to Microsoft SQL Server.** Only click this radio button if you already have SQL Server client software installed on this computer, and you plan to use MSSQL as your logging database.
   - **I need support for logging to Oracle.** Only choose this option if you have the Oracle client software installed on this computer, and you plan to use Oracle as your logging database server.

4. **Click the Next button.**

   The Choose Components window appears.

5. **In the Choose Components window, select the components you want to install and then click the Next button.**

   We recommend selecting all of the components. The Snort option is the snort executable, the Documentation option gives you a few documents on using Snort and the Contrib option installs the contrib directory containing goodies such as scripts for building database tables in the MySQL, MSSQL, PostGres, and Oracle database systems.

   The Install Location window appears.

6. **Choose a directory to install to.**

   We chose to keep all of our Snort-related applications in the same root directory on our D:\ drive (the data partition we mentioned). The path to our Snort installation is: D:\snortapps\snort, but you can install it anywhere on your drive.

7. **Click the Install button.**

8. **When the installation is complete, click the Close button.**

   An information window appears.

9. **Click the OK button.**

   You're done! Now it's time to move on to configuring your Snort system.

Bending Snort to Your Will

A new Snort installation requires a few configuration points. Conveniently, one file has all the configuration settings required (*Snortpath* is the path to your Snort installation):

```
Snortpath/etc/snort.conf
```

When you're ready to configure Snort, open snort.conf in a text editor. Figure 5-1 shows snort.conf in WordPad, but you can use:

- Edit (from the command line)
- Notepad
- Any other text editor that won't corrupt the text with crazy formatting characters the way some fully featured word processors will.

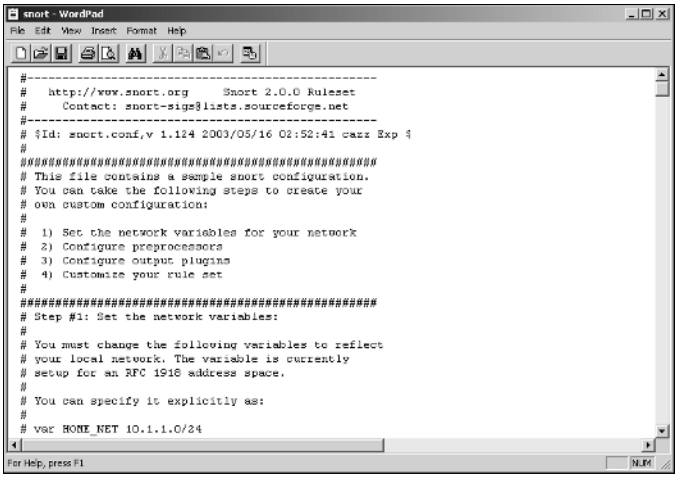

Figure 5-1: snort.conf is best viewed while firmly seated.

This configuration isn't a series of handy questions, button clicks, and good feelings. You're parsing through a flat text file and entering the proper settings by hand.

Double-check *everything* you type in to the snort.conf file. If entries aren't exactly correct, Snort doesn't work. Guaranteed.

The following configuration options in the snort.conf file are essential to a properly functioning Snort installation:

- Network settings
- Rules settings
- Output settings
- Include settings

Network settings

The network settings allow you to set Snort to monitor any range of network IP addresses, from a single IP address, several IP addresses in groups or individually, and entire IP subnets. You can configure the IP address range and the subnet.

The placement of the Snort sensor depends on both the configuration file and how much "pipe" it can suck from. In a switched environment, when using prodigious VLANs, additional network configuration may be required to give Snort the best possible sample of network traffic. Chapter 2 provides all the detail you need to set up Snort in any network environment.

You can control the network range that Snort monitors by changing the `var HOME_NET` setting in snort.conf. Your options are:

Entire network

By default, snort.conf contains the following line, which monitors the entire local network:

```
var HOME_NET any
```

If you don't change this setting, Snort monitors the entire network segment the Snort system is attached to.

Single IP address

To monitor a single IP or computer insert the IP address range and the subnet of the network or host into snort.conf. To do this, replace the existing var HOME_NET configuration line with this form:

```
var HOME_NET IPAddressRange/Subnet
```

The IPAddressRange/Subnet notation may not be something you're familiar with; it's not normally used to configure a network interface on Windows systems. This particular type of IP address notation is called CIDR notation, and we give you the run-down on it in Chapter 1, in the sidebar "Understanding CIDR notation."

The following examples monitor a Class C network with an IP address range of 192.168.10.0 – 192.168.10.255 and a subnet of 255.255.255.0:

- This line monitors the entire Class C network:

  ```
  var HOME_NET 192.168.10.0/24
  ```

- This line monitors a single host on the Class C network:

  ```
  var HOME_NET 192.168.10.2/32
  ```

Multiple hosts

You can specify a number of hosts within the network space you are monitoring by listing them in the `var HOME_NET` configuration statement. The line takes this form:

```
var HOME_NET IPAddressRange/Subnet,IPAddressRange/Subnet,...
```

Separate each IP address in the `var HOME_NET` configuration statement by a comma *without* spaces. If there are any blank spaces in the list of IP addresses and subnets, then Snort fails to start.

The following example monitors three hosts on a typical class C network:

```
var HOME_NET 192.168.10.2/32,192.168.10.3/32,192.168.10.6/32
```

Rules, rules, rules

So Snort can detect attacks and alert you when attacks occur, Snort needs to know where its rulebase is (and you need to know it if you want to write new rules).

By default, the rulebase is in *Snortpath*\rules (*Snortpath* is the location of the Snort install).

To set the rules path in the snort.conf file, replace the existing `var RULE_PATH` line with this form (*Snortpath* is the location of the Snort install):

```
var RULE_PATH SnortPath\rules
```

Figure 5-2 shows an example of a properly configured `var RULE_PATH` line in the snort.conf file.

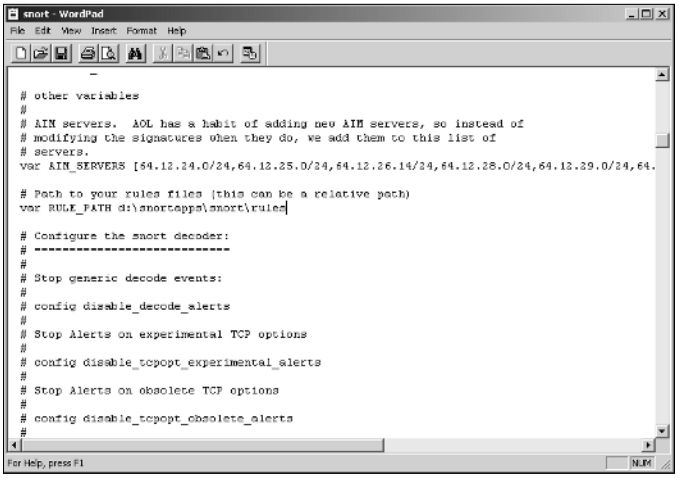

Figure 5-2: The Tao of Snort Rules.

Output settings

Output settings are very important in Snort, for they define how Snort's information will be presented to you. We go into output settings in-depth in Chapter 6, but for now we're concerned with configuring Snort to output to an alert text file and a database.

Alert output

The alert output setting is added to the snort.conf configuration file. The snort.conf file will also come in handy when we port that information into our MySQL database. Follow these steps:

1. **Find the output line that appears by default as:**

   ```
   # output log_tcpdump: tcpdump.log
   ```

   Because the default line begins with the comment character (#), Snort ignores it.

2. **Change the preceding default output line to this:**

   ```
   output alert_fast: alert.ids
   ```

   This setting creates a flat text file in the 'log' directory where Snort appends each alert created when one of its rules fires on incoming network packets.

Delete the comment character (#) from the beginning of the changed line so Snort doesn't ignore it when processing the configuration file.

Database outputs

These configuration settings configure Snort to push information to MySQL, the Windows database we recommend.

Even if MySQL hasn't been installed yet, this is the right time to get everything ready for MySQL on the Snort side of the house.

Collecting database information

Before configuring the database output settings, you must decide on the following information. Unless you're working with an existing database, these four settings are totally up to you.

Feel free to write your information in the following blanks, but guard it carefully or destroy it unless you want some wily social-engineering "3l33t haxor" to get all your database information.

- User: _____

  This is the MySQL user for the database where Snort stores its data. We like 'elvis' (who doesn't?), but it can be anything you want.

- Password: _____

  This is the password for the MySQL database user.

- dbname (for logs and alerts): _____

  This is the database name where Snort will store its alerts and logs.

- YOURHOSTNAME _____

  This is the hostname of your database server. If you are running your database on the same system as your Snort sensor, then it is the same name.

If you don't know your computer's hostname, you can find it by typing hostname at the command prompt. The prompt returns the hostname of your machine.

Don't use default users, database names, and passwords unless you want your box hacked.

Editing the output settings in snort.conf

When you have the database information ready, you can configure the output settings in `snort.conf`.

The following steps show how to edit `snort.conf` to log alerts to a MySQL database for your system. There are examples for our own test system, which has a MySQL database called `snorty` as the user `elvis` with a password of `3133th@x0R` on the local IP address (127.0.0.1) at port 3306 with a sensor name of `elvisisdead`.

If you plan to install your database on a separate server, put the correct IP address where the database resides. For this demonstration, the database is running on the same server as Snort.

Follow these steps to configure the output settings in the snort.conf file:

1. **Find the following default output line in the `snort.conf` file:**

```
# output database: log, mysql, user=root password=test
        dbname=db host=localhost
```

2. **Configure the logs. Using your own database information, change that default output line to something like this:**

```
output database: log, mysql, user=User password=Password
        dbname=dbname host=YOURHOSTNAME port=portnumber
        sensor_name=thesensorname
```

Delete # from the beginning of the changed line so Snort doesn't ignore the line.

For example, we changed the default to this line:

```
output database: log, mysql, user=elvis
        password=3l33th@x0R dbname=snorty host=127.0.0.1
        port=3306 sensor_name=elvisisdead
```

3. **Configure the alerts. Using your own database information, add a new output line like this:**

```
output database: alert, mysql, user=User
        password=Password dbname=dbname host=YOURHOSTNAME
        port=portnumber sensor_name=thesensorname
```

For example, we added this output line:

```
output database: alert, mysql, user=elvis
        password=3l33th@x0R dbname=snorty host=127.0.0.1
        port=3306 sensor_name=elvisisdead
```

Include configuration

Two standard Snort configuration files must be referenced for Snort to properly classify and provide references to the alerts it generates: classification.config and reference.config.

classification.config

classification.config holds alert levels for the rules that Snort monitors against network traffic.

To set the classification.config file in the snort.conf configuration file, follow these steps:

1. **Find this default line in the** `snort.conf` **file:**

```
Include classfication.config
```

2. **Insert the actual path for the** `classification.config` **file into the preceding** `Include` **line, like this:**

```
Include SnortPath\etc\classification.config
```

For example, the actual snort.conf file on our test system has this line:

```
Include D:\snortapps\Snort\etc\classification.config
```

reference.config

reference.config contains URLs referenced in the rules that provide more information about the alert event.

To set the `reference.config` file in the `snort.conf` file, follow these steps:

1. **Find this default line in the `snort.conf` file:**

   ```
   Include reference.config
   ```

2. **Insert the actual path for the `reference.config` file into the preceding `Include` line, like this:**

   ```
   Include SnortPath\etc\reference.config
   ```

   For example, the actual snort.conf file on our test system has this line:

   ```
   Include D:\snortapps\Snort\etc\reference.config
   ```

Testing the Installation

Snort runs in three different modes: Sniffer, Packet Logger, and Network Intrusion modes.

Sniffer mode

Sniffer mode is the simplest iteration of Snort. To run it, follow these steps:

1. **From the command line (within the `SnortPath\bin` directory) type**

   ```
   snort -v
   ```

   This command runs Snort as a packet sniffer with the `verbose` switch, outputting TCP/IP packet headers to the screen (see Figure 5-3). As you know if you're a coffee-guzzling network engineer, Snort is working at its most basic level. But don't panic . . .

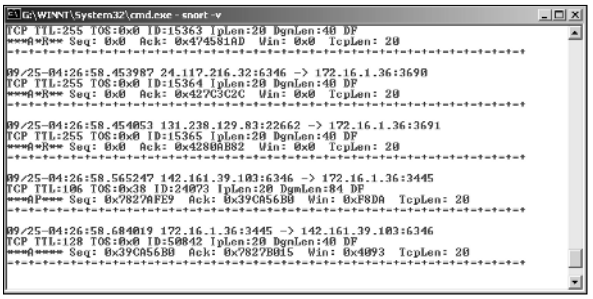

Figure 5-3: Aagh! What's that?

2. **Press Ctrl+C keys together to stop the output.**

   `Snort/WinPcap` summarizes its activities, as shown in Figure 5-4.

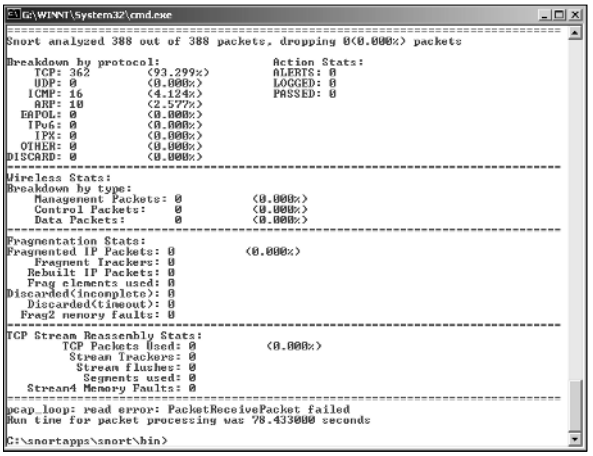

Figure 5-4: A summary of *Aagh! What's that?*

3. **To receive a more detailed capture of packets on the wire, type**

   ```
   snort -vd
   ```

   This command provides the TCP/IP headers and packet information (descriptive).

4. **Type `snort` at the command line for a full list of all the switches.**

   If you're getting TCP headers, you know that so far you're right on track.

5. **If you have more than one network card in your Snort IDS system, type**

   ```
   snort -W
   ```

   This command determines how `WinPcap` has these adapters numbered, and is only available in the Win32 version of Snort.

6. **If you're running Snort from the command line with two network adapters, specify which adapter to monitor:**

   ```
   snort -v -i#
   ```

   # is the number of the applicable adapters (as shown on the output of the `snort -W` command).

You must use this `-i` switch whenever you run the `snort` program on the command line.

Packet Logger

You can test Snort's logging abilities with the `-l (log)` switch, by typing:

```
snort -dev -l SnortPath\log
```

This runs Snort in descriptive verbose mode and logs all its findings to the directory called log under the Snort install directory. The individual packets are filed in hierarchical directories based on the IP address from where the packet was received, as seen in Figure 5-5.

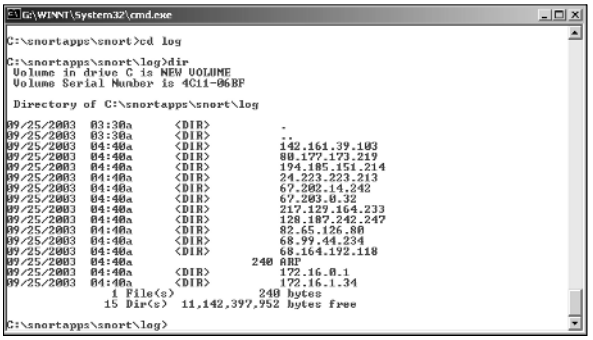

Figure 5-5: Pretty cool-looking, but not a very useful way to do it.

Several command-line switches are specific to logging and output, including the ability to log all packets to a single binary file. Play around with those as needed. Chapter 6 goes over a few of your options.

Setting Up MySQL for Snort

While MySQL isn't required with Snort, it is required for a front-end console such as ACID. If you set up MySQL or another database system, you can see the alerts without the front-end console, but you really don't need that kind of pain.

Installing MySQL

Before you install MySQL, you have to get hold of it. MySQL can be downloaded from `http://www.mysql.com/downloads/index.html`.

Get the *latest* production version of MySQL for your Windows operating system.

When you've downloaded, perform these steps to install MySQL:

1. **Uncompress the MySQL ZIP file into a temporary directory.**

   This file is ZIP file usually called something like `mysql-4.0.17-win.zip`. You need a compression utility (such as WinZip or WinRAR) to uncompress it on a Windows 2000 platform, but Windows XP has built-in transparent access to compressed archives with extensions such as `zip`, `gzip` and `tar`.

2. **Where you uncompressed the file, double-click `setup.exe`.**

   The Welcome window appears.

3. **Click Next, read the information, and click Next again.**

   The Information window appears. If you install MySQL in a directory other than `C:\mysql`, you must create an initialization file; the Information window describes this process.

4. **At the Destination Location window, click Next if you want to install it to the default directory (`c:\mysql`).**

5. **Choose the Typical install and click Next.**

   MySQL installs itself.

6. **When the installation is finished, click the Finished button.**

   You're all installed now.

To finish the initial configuration of MySQL, perform these steps:

1. **Open a command window and navigate to**

   `$`*`SQLPATH`*`\bin`

   `$SQLPATH` is the path to the directory in which you've installed `mysql` (ours is `C:\mysql\`).

2. **In the *SQLPath*\bin directory, type the following command:**

   ```
   winmysqladmin
   ```

   The MySQL administration console window appears and prompts you for a login.

Whoa, red light!

If the MySQL Admin traffic light is Red, MySQL can't start. It probably can't read its `.cnf` and `.ini` files. MySQL first reads the `my.ini` file (usually located in the `C:\Winnt` directory). If it can't read that, it reads the `my.cnf` file, usually located in the root directory (`C:\`). Check both of those files with a text editor (WordPad or Notepad) and ensure that lines like these appear (carefully check the slashes):

```
basedir=SQLPath
datadir=SQLPath/data
 [WinMySQLadmin]
Server=SQLPath/bin/mysqld-
   nt.exe
user=USER
password=PASS
```

For the preceding lines, the variables in your files should have these values:

- `SQLPath`: the path to where you installed MySQL (that is, the `root` MySQL directory path)
- `USER`: the MySQL account's username
- `PASS`: the MySQL account's password

The resulting configuration file looks something like this:

```
basedir=C:/mysql
datadir=C:/mysql/data
 [WinMySQLadmin]
Server=C:/mysql/bin/mysqld-
   nt.exe
user=root
password=Fry4tat3rZ
```

Carefully check the orientation of the slashes (`/`). They are Unix-like forward slashes, not the backslashes you typically use with the Windows filesystem. If these aren't right, MySQL won't start.

3. **Use any login name and password you want.**

   This sets the `root` password for MySQL. Ours looked like this:

   ```
   login: root
   password: Fry4tat3rZ
   ```

4. **Click the OK button, and MySQL starts up as a service.**

   A traffic light appears in your system tray, showing a green light.

   If the light is red, MySQL can't start. The "Whoa, red light!" sidebar in this chapter can help you diagnose the problem.

Configuring MySQL for Snort

When MySQL is up and running, you're ready to configure it to take data from Snort. To check that everything is A-OK, right-click the traffic-light icon in the system tray and click Show Me. Click the Start Check tab. The `my.ini` line should show a `yes`, and all other lines should show `OK`, as in Figure 5-6.

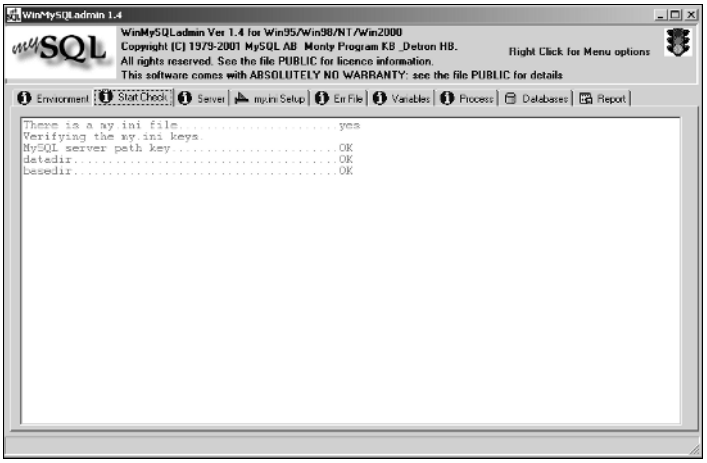

Figure 5-6: Everything is A-OK with MySQL.

Setting up the my.ini file

You can set up the my.ini file from the Admin console (`winmysqladmin`) or with a text editor. We prefer the Admin console; you can edit the `my.ini` file directly from the Setup tab of the `winmysqladmin` console. To set up the `my.ini` file using the Admin console, perform the following steps:

1. **Run `winmysqladmin` from a command prompt.**
2. **Bind MySQL to this system's localhost IP address.**

   In this case, it's `127.0.0.1`.
3. **Set the communication port.**

   For a typical MySQL installation, it's 3306.
4. **Set the `key_buffer` setting for Snort data.**

   (We chose to keep no more than 64MB in the Snort buffer.) When you're finished, the text in the `my.ini` Setup tab should look like the following code snippet (with the exception of the password line, which should contain your password):

```
#This File was made using the WinMySQLAdmin 1.4 Tool

#Uncomment or Add only the keys that you know how works.
#Read the MySQL Manual for instructions

[mysqld]
basedir=C:/mysql
bind-address=127.0.0.1
datadir=C:/mysql/data
#language=C:/mysql/share/your language directory
#slow query log#=
#tmpdir#=
```

```
port=3306
set-variable=key_buffer=64M
[WinMySQLadmin]
Server=C:/mysql/bin/mysqld-nt.exe
user=root
password=YOURPASSWORD
```

Any text that follows a pound (#) symbol is a comment and ignored by MySQL. If your code looks the way it should, then your `my.ini` file is set.

5. **Click the Save Modifications button (in the lower-left corner).**

   MySQL prompts you to confirm the changes.

6. **Click the Yes button.**

   MySQL alerts you that the changes have been made and confirmed.

7. **Click the OK button.**

   The changes are accepted.

8. **Right-click anywhere on the window, and then click the Hide Me menu option to close the console.**

Digging in SQL guts

Once MySQL is configured properly, clean up MySQL, configure it for Snort, and secure it. All this essential stuff is done from the command line.

In the MySQL command interface, every command must end in a semicolon (;).

The first order of business is to clean up some chaff by deleting the default databases.

1. **In the *SQLPath*\bin directory, log in to `mysql` from the Windows command prompt, type the following command, and press Enter:**

   ```
   mysql -u root -p
   ```

   You're asked for your MySQL `root` password.

2. **Enter the `root` password and press Enter.**

   A welcome message reminds you that commands must end with a semicolon. Your prompt changes to

   ```
   mysql>
   ```

3. **At the prompt, type the following and then press the Enter key:**

   ```
   use mysql;
   ```

   This command puts you in the database called `mysql`.

4. **Get rid of any `host` entries, like this:**

```
delete from user where host = "%";
```

5. **Delete other user accounts, like this:**

```
delete from user where user = "";
```

6. **Make sure the `root` account is the only user account here, like this:**

```
select * from user;
```

   This command displays user information. You should only see `root` as a user.

7. **Delete the `test` database by typing the following:**

```
drop database test;
```

8. **Ensure that only the `mysql` database exists by typing this command:**

```
show databases;
```

   The following should appear:

```
+----------+
| Database |
+----------+
| mysql    |
+----------+
1 row in set (0.00 sec)
```

   If you get that, you're ready to create your Snort databases.

Create the Snort databases

At the `mysql>` prompt, type the following commands and press the Enter key after each one:

```
create database snort;
create database archive;
```

When you execute a `show databases;` command now, you should see this:

```
+----------+
| Database |
+----------+
| mysql    |
| snort    |
| archive  |
+----------+
1 row in set (0.00 sec)
```

Creating Snort's user accounts

With the Snort databases in place, set up the user accounts that Snort is to use when it logs in to add data to its databases. As an example, the following steps walk through setting up the `elvis` user account.

1. **At the `mysql>` prompt, type the following and press Enter after each line:**

   ```
   grant INSERT,SELECT,UPDATE on snort.* to elvis@localhost
           identified by "3l33th@xOR";
   ```

   Refer to Chapter 7 for more about user setup.

2. **Verify the `elvis` user's permissions type:**

   ```
   show grants for elvis@localhost;
   ```

   MySQL displays the `elvis` user's permissions, which should match those you gave the `elvis` user account when you created it.

3. **If you made a mistake, go back and redo the `snort` user account's permission.**

   The `snort` user account must be allowed to do its business, otherwise nothing will work.

Is this thing on?

After configuring the user accounts, make sure everything's working:

1. **Open the MySQL console by issuing the following command:**

   ```
   winmysqladmin
   ```

2. **Click the `my.ini` Setup tab, and then click the Create ShortCut on Start Menu button.**

   This creates a shortcut in the Windows Startup folder so MySQL starts automatically when your Windows 2000 box starts.

3. **Check the Windows Task Manager (right-click the Windows toolbar and select Task Manager).**

   In Windows 2000 and XP, you can press Ctrl+Shift+Esc instead.

   Here's where you make sure that both `snort.exe` and `mysqld-nt.exe` are running in the Process list. If you are still running the console, `winmysqladmin.exe` also appears in the process list.

Locking MySQL and throwing away the key

Choose a strong password for your `root` user and make sure you remember it, since it's your only admin interface to MySQL. *Don't* duplicate our example (`Fry4tat3rZ`) — but you knew that. Choose a password that includes numbers, upper- and lowercase letters, and special characters. To change the password for the `root` user, type the following at the `mysql>` prompt and press the Enter key:

```
set password for  = password ("YOURPASSWORDHERE");
```

You should get a confirmation. Then you can type `quit;` and press the Enter key to exit.

Configuring Snort as a Service

To run Snort as a background service on Windows 2000, XP, or 2003, you must know

- Where your rules directory is
- Where you want Snort to create its log file

When we added the database output configuration to the `snort.conf` file, we made Snort rely on MySQL. If we try to run Snort as a service without having MySQL installed and configured, the Snort service fails because it's looking for MySQL databases. Keep this in mind if you configured Snort for MySQL support, but skipped the section on installing and configuring MySQL.

The following examples are a generic configuration. Your configuration may vary slightly.

Windows 2000, XP, and 2003 service commands

The general procedures for installing and uninstalling services on a Windows 2000 system are pretty straightforward:

- To install a program as a service on Windows 2000, XP, or 2003, execute the following command at the command line (replace *`Program`* with the executable you want to install as a service):

  ```
  Program /SERVICE /INSTALL
  ```

- To Uninstall a program from the Services, execute this command (replace *Program* with the executable you want to uninstall from the Services):

```
Program /SERVICE /UNINSTALL
```

Installing Snort as a service

To install Snort as a service, follow these steps:

1. **Specify your Snort path by typing the following command at the command line (in the `/bin` directory of your Snort installation) and then pressing Enter:**

```
snort /SERVICE /INSTALL -de -c SnortPath\etc\snort.conf
        -l SnortPath\log -i#
```

   For the preceding command:

   - `snort` is the name of the Snort executable.
   - `/SERVICE` is the Windows command to access the Services commands.
   - `/INSTALL` is the Services command that installs the program as a Windows service.
   - `-de` is a pair of switches: the `-d` switch tells Snort to dump Application-Layer network information; the `-e` switch displays Second-Layer header information.
   - `-c $SnortPath\etc\snort.conf` is where the `-c` switch tells Snort to use the configuration file specified by `$SnortPath\etc\snort.conf`.
   - `-l $SnortPath\log` is where the `-l` switch (that's a lowercased *L*) tells Snort to log to the path: `$SnortPath\log`.
   - `-i#` tells the `-i` switch tells Snort to capture log data on the network interface specified, and # is the number of the interface you want Snort to monitor.

     (If you're unsure which network adapter you want Snort to monitor, type `snort -W` to list available interfaces — and then choose one. Note that the switch is case-sensitive.)

     

   - *SnortPath* is the path to your `root` installation of Snort. (For example, ours is `D:\snortapps\snort`.) The # sign after the `-i` switch represents the actual network interface that you want Snort to monitor.

2. **Specify the `-i` switch by typing the following command at the command line and then pressing Enter:**

```
snort -W
```

Using WinpCap, Snort outputs the names of your network adapters (probably just one) to the screen, preceding each one by a number. This number is the one you want for the `-i` switch.

If the service fails to start or if you get an error after executing this command, make sure that you

- Typed the command correctly
- Properly configured the `snort.conf` file (as discussed in the preceding sections)

In some situations, you won't receive a specific failure message for Snort. In these cases, check the Event Viewer in the Windows Control Panel for details about the error. Usually, there are problems with either your `snort.conf` file or your MySQL (or other database system) installation.