Interested in learning more about security?

SANS Institute InfoSec Reading Room

Distributed NIDS: A HOW-TO Guide

Network Intrusion Detection Systems (NIDSs) can be an invaluable tool if implemented and used effectively. A properly configured system can be a great utility for monitoring and tracking network activity, watching network trends and observing behavioral patterns. However, the design, installation, configuration and monitoring of an NIDS can be a very daunting task. This goal of this guide is to cover these four components, and to leave the reader with a fully functional and powerful distributed NIDS as a result.

Distributed NIDS: A HOW-TO Guide

Alan McCarty

September 4, 2003

Table of Contents

Introduction

Network Intrusion Detection Systems (NIDSs) can be an invaluable tool if implemented and used effectively. A properly configured system can be a great utility for monitoring and tracking network activity, watching network trends and observing behavioral patterns. However, the design, installation, configuration and monitoring of an NIDS can be a very daunting task. This goal of this guide is to cover these four components, and to leave the reader with a fully functional and powerful distributed NIDS as a result.

The guide is laid out in two parts. The first part will cover the design of the system, explaining the logical layout of a distributed NIDS and applying the layout to a fictional corporate network. The first part will also detail the gathering of required hardware and software. The second part will explain the building of the system by leading the reader through the remaining components and observation and supervision steps.

While the intended audience of this guide is system and network administrators, it is written so that anyone with an intermediate knowledge of computers and networks should be able to understand the concepts and follow the steps to build a complete system.

Part I

Design

For those of you who are familiar with NIDSs, this next bit may be a review. The goal of IDS, (or NIDS as we'll refer to it), is to monitor network assets to detect anomalous behavior and misuse.[1] While many other features and capabilities may be lumped into a broader definition of IDS, this explanation clearly states what an IDS does.

Through the use of standard PC hardware and open-source[*] software, this guide will add to the definition above to include the analysis, correlation and presentation of collected data.

This portion of the guide focuses on where and why to place certain pieces of our distributed NIDS. Before we go any further, we need to define the pieces that make up the system. Our distributed NIDS consists of sensors and an engine. Sensors are the devices that reside on various network segments, listening to or "sniffing" that segment's traffic. The sensors are configured with intrusion detection software and a database client. The sole purpose of a sensor is to monitor every packet on its network segment, and report back to the engine if

[1] Innella
[*] http://www.opensource.org/

any packets look suspicious. The engine's purpose is three-fold. Its main function is to serve as a database server, collecting data from the sensors. Its secondary function is that of a "looking glass" into the database and it's collected data. The engine's final task is to manage the sensors to ensure that they're watching for the right traffic, using the most updated configurations, and operating appropriately. Figure 1-1 below illustrates the logical and functional layouts of the sensors and engine.

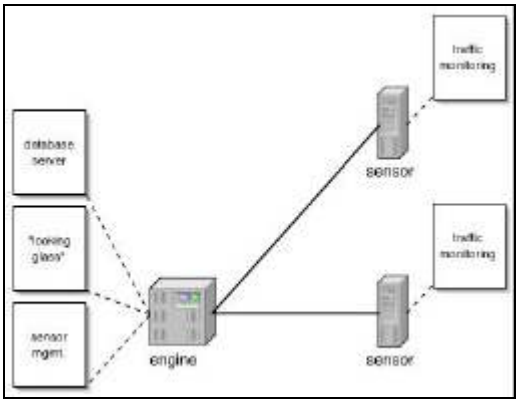


Figure 1-1

Now that we know what the pieces of our NIDS do, we need to figure out where to put them. For simplicity's sake, let's assume a fictitious corporate network that contains two segments; one public and one private, separated by a firewall (see Figure 1-2). Since we only have two segments, we should monitor both of them, as threats can originate on either side of the firewall. It's most likely that the higher volume of network-based threats will come from the Internet, so we need the sensor on the public segment to be less sensitive to false positives. On the other side of the firewall, the sensors need to be extremely sensitive. Any malicious traffic on this segment is a higher threat to assets on the private network. Armed with this knowledge, we can determine the immediate need for two sensors; one to monitor each important area of our network.

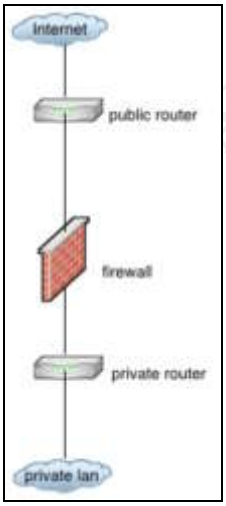


Figure 1-2

If we combine our logical layout with our fictitious network, we get a layout that looks something like Figure 1-3, below. We now know where we want our sensors, but we need to determine which network segment on which to put the engine. Since we don't want it publicly accessible, putting it on the public segment is out of the question. We can put it on the private segment, but it won't be able to communicate with the public sensor.

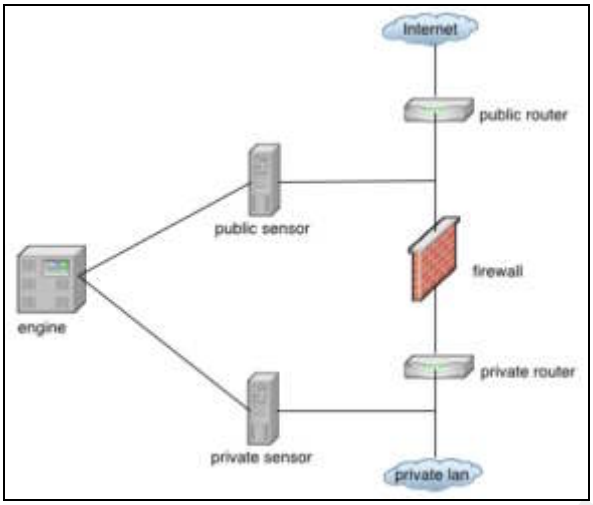


Figure 1-3

A solution is to create a second private network, to be used strictly for NIDS and other types of network management. By dual-homing our sensors (see Figure 1-4), we create a "back-end" network that will allow us to access and manage our engine and sensors on the same network. In addition, we'll dual-home our engine back to the private LAN so that we can access our management network from there.

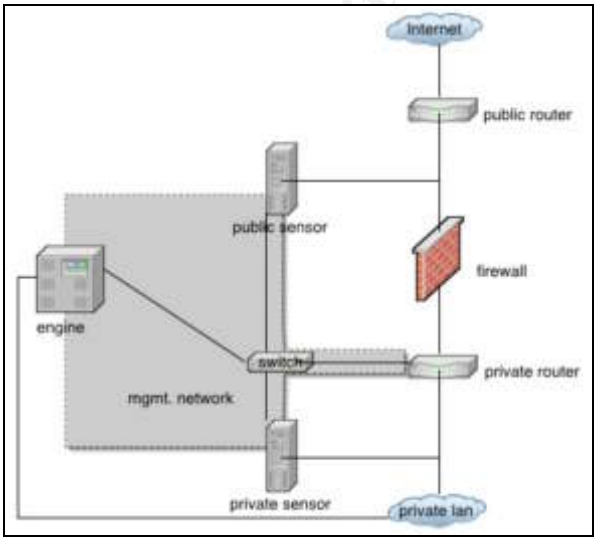


Figure 1-4

Even though we've separated our management traffic out, we still need to ensure that the sensors themselves are bridging the public and private networks by way of the management network. Keeping security foremost in our system, we'll address this "hole" later in the guide.

IP Addressing

For our management network, we'll use an RFC 1918 class C network of 192.168.100/24. This will give us plenty of room to add new engines and sensors in the future. Figure 1-5 illustrates the subnets and addresses we'll use. In addition, the private and public networks have been assigned subnets.

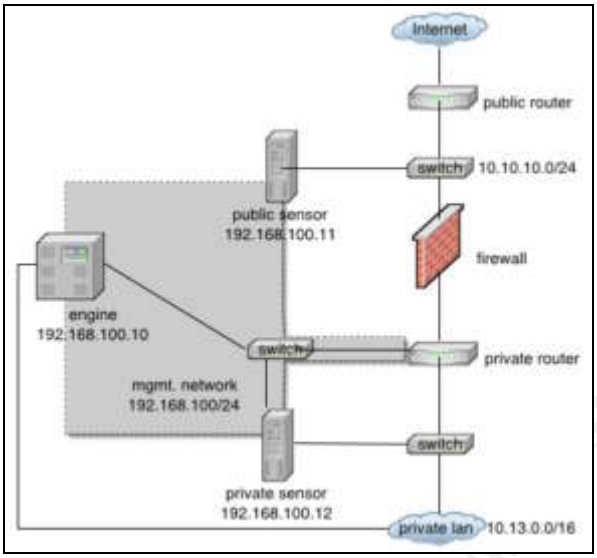


Figure 1-5

Required Resources

The physical pieces of our distributed NIDS can be further separated into hardware and software components. This portion of the guide will introduce these elements and lead directly into the next section, in which we'll build and configure the engine and sensors.

Hardware

Assuming the network hardware in our fictitious corporate network are already in place and working, we'll cover what components we need to add to the system, and any changes we need to make. The first thing we need to do is ensure that we have a way to monitor or "sniff" each network segment. Figure 1-4 illustrated that each sensor intersects its segment with a line. In order for us to properly monitor network traffic, we need to add a switch to each segment at that intersection. (see Figure 1-5) We could add a hub, but if our network segments are passing large amounts of traffic, a hub's susceptibility to collisions will adversely affect the network, as well as the sensor's ability to process all of the packets. On the other hand, if it's the only available option, a hub will have to do.

While a switch is preferable, we need a switch that has management capabilities. In particular, the switch needs to allow us to establish a monitor port, also known as a mirror or span port. Enabling this feature on the switch will instruct the switch to send all packets to the monitor port, as well as to the packets' intended destinations. If we then plug our sensors into the monitor port on these switches, we'll be able to sniff all of the traffic on the switch. Standard CAT5 cabling should be used for all sensor and engine connections.

Now, we'll discuss the sensors and engine hardware. As stated in the beginning of the guide, we'll be using standard PC hardware and open source software. The main justification for this approach is cost. Complete PC systems can be bought or built inexpensively, and when coupled with open source operating systems and software can be extremely fast and powerful. This means that additional sensors can be added quickly and affordably when needed.
Since the engine and sensors perform different tasks, they may require subtle differences in hardware. For example, the engine will be gathering, organizing and storing large amounts of data, so it may be a good idea to allocate more disk space to the engine than to the sensors. In addition, the engine will be running multiple processes for its three functions (database, looking-glass and management), so it may require more RAM than the sensors. For the engine and sensors, 256MB of RAM is a good place to start, though 512MB or more is preferred. Since RAM is such an inexpensive commodity these days, it's affordable and easy to add more. The sensors will be gathering data from the networks at very high rates, and often capturing it from multiple data streams simultaneously. For this, they may need a faster CPU and network interface card (NIC). A CPU equivalent to an Intel Pentium III 500mhz should be sufficient for most sensors, though faster is always better. Again, with PC components as inexpensive as they are today, it's easy to fill RAM slots to their capacity.
At a minimum, the sensors' NIC should run as fast as or faster than those of the other hosts on the switch. This way you can be sure to capture as much data as possible. Below is a summary of the hardware necessary for our NIDS: [*]

- Two manageable switches (or hubs) for sniffing public and private segments
- One switch (or hub) for the management network
- Six standard CAT5 cables
- Two sensor servers
- One engine server

Software
Since monetary costs are critical to most of us, we'll be taking advantage of many open source software packages that are very affordable, if not free altogether. Table 1-1 displays a functional summary of the software we'll be using:

[*] As a reference, a list of the hardware used in a test lab when writing this guide is available in Appendix III.

Table 1-1

Engine:
- Operating System: Red Hat Linux 9.0
- Database Server: MySQL
- Sensor Management: SnortCenter Management Console
- "Looking Glass": ACID
- Supplemental Packages
  - o PHP – a general-purpose scripting language used for web development
  - o ADOdb
  - o Jpgraph

Sensors:
- Operating System: Red Hat Linux 9.0
- Intrusion Detection System: Snort
- Database Client: MySQL (client)
- Sensor Management Client: SnortCenter Agent
- Supplemental Packages
  - o NetSSLeay

Those of you who are familiar with various Linux distributions may have a preference for one other than Red Hat. If this is the case, feel free to substitute your preferred distribution for Red Hat for the remainder of this guide. While there may be some discrepancies during the configuration, the overall procedure should work fairly well with other Linux distributions.
Red Hat 9.0 can be downloaded in ISO image format from FTP mirror sites located at the following URL:

http://www.Red Hat.com/download/mirror.html

The suggested method of obtaining the additional required packages is to download them as well. URLs for the necessary packages are shown in Table 1-2.

Table 1-2

Snort v. 2.01 http://www.snort.org/dl
SnortCenter 1.0-RC1: http://users.pandora.be/larc/download/
SnortCenter Agent 1.0-RC1: http://users.pandora.be/larc/download/-ADOdb v. 3.50: http://php.weblogs.com/ADODB
ACID v. 0.9.6b23: http://www.andrew.cmu.edu/~rdanyliw/snort/snortacid.html
JpGraph v. 1.12.2: http://www.aditus.nu/jpgraph/jpdownload.php
Net::SSLeay v. 1.23 http://dag.wieers.com/packages/perl-Net-SSLeay
MySQL v. 3.23 or higher: http://www.mysql.com/downloads/index.html (included in Red Hat

installation)
PHP v. 4.0 or higher: http://www.php.net/downloads.php (included in Red Hat 9 installation)

Once all of the software has been downloaded, it is suggested that you put the packages on a CD or other removable device. Since our sensors will not be able to reach the Internet, having the packages available by other means is important. This is also the recommended method for installing updates and patches to the software and OS.

Part II

Installation

This portion of the guide details the steps for installing and configuring our NIDS. In the next section, we'll install the required packages, and configure our engine and sensors. For those of you who are unfamiliar with installing Linux, Appendices I and II detail a minimal Red Hat 9 installation for both an engine and a sensor. These may be helpful if you've never installed Red Hat before, or would like to perform a minimal installation. During the configuration and setup steps, there may be some commands that are foreign to you. If this is the case, it's suggested that you read the manual pages[*] for these commands so that you have an idea of what the command is doing.

Use Table 1-3 as a reference for the engine and sensor IP addressing. These addresses are used during the Red Hat 9 installation as well.

Table 1-3

Engine:(eth0)
IP Address: 192.168.100.10
Subnet Mask: 255.255.255.0
Gateway Address: (not needed)[**]
Engine: (eth1)
IP Address: 10.13.10.101
Subnet Mask: 255.255.0.0
Gateway Address: 10.13.0.1
Public Sensor:(eth0)
IP Address: 192.168.100.11
Subnet Mask: 255.255.255.0
Gateway Address: (not needed)
Public Sensor:(eth1)
IP Address: 1.1.1.1 (we'll change this later)
Subnet Mask: 1.1.1.1 (we'll change this later)
Private Sensor:(eth0)
IP Address: 192.168.100.12
Subnet Mask: 255.255.255.0
Gateway Address: (not needed)
Private Sensor:(eth1)
IP Address: 1.1.1.1 (we'll change this later)
Subnet Mask: 1.1.1.1 (we'll change this later)

[*]See http://www.tldp.org

[**]We don't need a gateway address since neither the engine nor sensors need to access the private LAN or the Internet. However, if you'd like to add one, feel free to do so.

Engine Installation

NOTE:
The sections below follow a simple convention. If words are contained in a box like this, then they are actual commands (often followed by results of the command) that need to be typed.

In addition, if a step is optional, its title will *appear in italics, like this*. This means that the step isn't required for functionality, but has been added as an optional feature.

1. Basic engine preparation*

1.1 Turn off unnecessary services.
- Logon as root

```
[root@ids-eng-01 pkg]#  chkconfig --list | grep :on
                kudzu          0:off   1:off   2:off   3:on    4:on    5:on    6:off
                syslog         0:off   1:off   2:on    3:on    4:on    5:on    6:off
                netfs          0:off   1:off   2:off   3:on    4:on    5:on    6:off
                network        0:off   1:off   2:on    3:on    4:on    5:on    6:off
                random         0:off   1:off   2:on    3:on    4:on    5:on    6:off
                rawdevices     0:off   1:off   2:off   3:on    4:on    5:on    6:off
                keytable       0:off   1:on    2:on    3:on    4:on    5:on    6:off
                apmd           0:off   1:off   2:on    3:on    4:on    5:on    6:off
                atd            0:off   1:off   2:off   3:on    4:on    5:on    6:off
                gpm            0:off   1:off   2:on    3:on    4:on    5:on    6:off
                iptables       0:off   1:off   2:on    3:on    4:on    5:on    6:off
                sshd           0:off   1:off   2:on    3:on    4:on    5:on    6:off
                sendmail       0:off   1:off   2:on    3:on    4:on    5:on    6:off
                rhnsd          0:off   1:off   2:off   3:on    4:on    5:on    6:off
                crond          0:off   1:off   2:on    3:on    4:on    5:on    6:off
                anacron        0:off   1:off   2:on    3:on    4:on    5:on    6:off
[root@ids-eng-01 /]# chkconfig --level 0123456 keytable off
[root@ids-eng-01 /]# chkconfig --level 0123456 kudzu off
[root@ids-eng-01 /]# chkconfig --level 0123456 sendmail off
[root@ids-eng-01 /]# chkconfig --level 0123456 netfs off
```

1.2 Add a non-root user so we're not logging into the box as root.

```
[root@ids-eng-01 /]# useradd admin
[root@ids-eng-01 /]# passwd admin
```

* Section 1 should not be considered a complete Red Hat hardening guide. Additional resources should be used to further secure the operating system.

```
Changing password for user admin.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

1.3 Add the non-root user to the sudoers list.

```
[root@ids-eng-01 /]# vi /etc/sudoers
```

- Add: admin ALL=(ALL) ALL

- Save the file

1.4 Log off of root account, log back on as admin.

1.5 Turn off sshv1 and prevent root logon over ssh. If this is the first time the sudo command has been invoked, a quick lecture will be displayed, and you will be prompted for the admin account password.

```
[admin@ids-eng-01 admin]$ sudo vi /etc/ssh/sshd_config
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

        #1) Respect the privacy of others.
        #2) Think before you type.
        #3) With great power comes great responsibility.

Password: <enter admin password> (keep this in a safe place)
```

- Change: #Protocol 1,2
  To: Protocol 2

- Change: #PermitRootLogin yes
  To: PermitRootLogin no

- Save the file

1.6 Replace /etc/motd with the text in Table 1-4, or something similar: (include leading and trailing blank lines)

Table 1-4

```
**************************************************************************
        NOTICE TO USERS

        This computer system is for authorized use only. Users (authorized or
```

unauthorized) have no explicit or implicit expectation of privacy.

Any or all uses of this system and all files on this system may be intercepted, monitored, recorded, copied, audited, inspected, and disclosed to authorized site and law enforcement personnel.

By using this system, the user consents to such interception, monitoring, recording, copying, auditing, inspection, and disclosure at the discretion of authorized site.

Unauthorized or improper use of this system may result in administrative disciplinary action and civil and criminal penalties. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. LOG OFF IMMEDIATELY if you do not agree to the conditions stated in this warning.

****

1.7 Create /etc/hosts entries for sensors. Since we may or may not be using DNS servers, we'll add host entries for our sensors so that their names appear in log files, etc., rather than their IP addresses. If you have access to a DNS server, you can add its IP address to /etc/resolv.conf.

```
[admin@ids-eng-01 admin]$ sudo vi /etc/hosts
```

- Edit the third line to read:

```
127.0.0.1       localhost.localdomain localhost
```

- Add the following lines:

```
192.168.100.10      ids-eng-01
192.168.100.11      ids-public-01
192.168.100.12      ids-private-01
```

- Save the file

1.8 Create a folder in admin's home (/home/admin) folder called "packages", and copy the following files from your removable media into that folder:

acid-0.9.6b23.tar.gz
adodb370.tgz
create_mysql
jpgraph-1.12.1.tar.gz
snortcenter-v1.0-RC1.tar.gz
snort-2.0.1.tar.gz

1.9 Reboot the server for our changes to take affect.

2. Setup MySQL and the Snort database.

2.1 Configure mysqld to start on boot.

```
[admin@ids-eng-01 admin]$ sudo /sbin/chkconfig --level 3 mysqld on
```

2.2 Move mysql home to separate partition.
If you used Appendix II as a guide for installing Red Hat 9, you created a /db partition.If you didn't, and you have a separate partition where you'd like to house your database, (rather than /var/lib/mysql) you can do that as well. Use the commands below to link /var/lib/mysql to /db/mysql.

- Stop mysqld if it is running

```
[admin@ids-eng-01 admin]$ sudo /etc/init.d/mysqld stop
```

- Copy /var/lib/mysql /db/mysql

```
[admin@ids-eng-01 admin]$ sudo cp -R /var/lib/mysql /db/
```

- Move /var/lib/mysql to /var/lib/mysqlOLD

```
[admin@ids-eng-01 admin]$ sudo mv /var/lib/mysql /var/lib/mysqlOLD
```

- Create a symlink at /var/lib/mysql to /db/mysql

```
[admin@ids-eng-01 admin]$ sudo ln -s /db/mysql /var/lib/mysql
```

- Restart mysqld

```
[admin@ids-eng-01 admin]$ /etc/init.d/mysqld start
```

2.3 Connect to the mysql server and create snort database.

```
[admin@ids-eng-01 admin]$ mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 3.23.54

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

- Set the password for the root user

```
mysql> set password for 'root'@'localhost' = password('<enter password here>');
```

- Create the snort database

```
mysql> create database snort;
```

- Exit mysql

```
mysql> exit
Bye
```

2.4 Create snort database schema and create snort user.

- Decompress snort-2.0.1.tar.gz to use the schema file

```
 [admin@ids-eng-01 admin]$ tar xvf /home/admin/packages/snort-2.0.1.tar.gz -C
/home/admin/packages
```

- Connect to mysql

```
[admin@ids-eng-01 admin]$ mysql -u root -p
Enter password: <password from step 2.3>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 3.23.54

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

- Connect to the snort database

```
mysql> connect snort;
```

- Import the schema from /home/admin/packages/snort-2.0.1/contrib

```
mysql> source /home/admin/packages/snort-2.0.1/contrib/create_mysql
        Query OK, 0 rows affected (0.01 sec)
        Query OK, 1 row affected (0.01 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.01 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
```

```
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.02 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.01 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
mysql>
```

- Assign privileges within the snort database to the snort user

```
mysql> grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.* to
snort;
Query OK, 0 rows affected (0.01 sec)

mysql> grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.* to
snort@localhost;
Query OK, 0 rows affected (0.00 sec)
mysql>
```

- Connect to the mysql database

```
mysql> connect mysql
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Connection id:    4
Current database: mysql

mysql>
```

- Assign a password to the snort user

```
mysql> set password for 'snort'@'localhost' = password('<enter password
here>');
Query OK, 0 rows affected (0.00 sec)
mysql> set password for 'snort'@'' = password('<enter password here>');
Query OK, 0 rows affected (0.00 sec)
```

- Exit mysql

```
mysql> exit
```

```
Bye
```

3. Install and configure ACID and supporting packages.

3.1 Install ACID and supporting packages. ACID, as well as its supporting packages, "live" inside the root apache document folder. In the case of Red Hat 9, the root is /var/www/html. Other Linux distributions may vary.

- Decompress ACID from /home/admin/packages/acid-0.9.6b23.tar.gz into /var/www/html/acid

```
[admin@ids-eng-01 admin]$ sudo tar xvzf /home/admin/packages/acid-0.9.6b23.tar.gz -C /var/www/html/
```

- Decompress ADOdb from /home/admin/packages/adodb370.tgz into /var/www/html/adodb

```
[admin@ids-eng-01 admin]$ sudo tar xvzf /home/admin/packages/adodb370.tgz -C /var/www/html/
```

- Decompress JpGraph from /home/admin/packages/jpgraph-1.12.1.tar.gz to /home/admin/packages/jpgraph-1.12.1

```
[admin@ids-eng-01 admin]$ cd /home/admin/packages/
[admin@ids-eng-01 admin]$ tar xvzf jpgraph-1.12.1.tar.gz
```

- Rename and move /home/admin/jpgraph-1.12.1 to /var/www/html/jpgraph

```
[admin@ids-eng-01 admin]$ sudo mv /home/admin/packages/jpgraph-1.12.1 /var/www/html/jpgraph
```

3.2 Configure ACID.

- Edit the ACID configuration file

```
[admin@ids-eng-01 admin]$ sudo vi /var/www/html/acid/acid_conf.php
```

- Change: $DBlib_path = "";
  To: $DBlib_path = "../adodb";

- Change: $alert_dbname = "snort_log";
  To: $alert_dbname = "snort";

- Change: $alert_password = "mypassword";
  To: $alert_password = "<password from step 2.3>";

- Change: $ChartLib_path = "";
  To: $ChartLib_path = "../jpgraph";

- Save the file

3.2 Configure Apache.

- Edit the apache configuration file

```
[admin@ids-eng-01 admin]$ sudo vi /etc/httpd/conf/httpd.conf
```

- Add the lines from Table 1-5 after the section that starts with **“ #<Directory /home/*/public_html>** "

Table 1-5

```
<Directory "/var/www/html/acid">
        AuthType Basic
        AuthName "ACID Console Admin"
        AuthUserFile /usr/lib/apache/passwords/passwords
        Require user admin
        AllowOverride None
</Directory>
```

- Change: #ServerName new.host.name:80
  To: ServerName ids-eng-01

- Save the file

3.3 Set a password for the ACID web pages.

- Create the /usr/lib/apache and /usr/lib/apache/passwords folders

```
[admin@ids-eng-01 admin]$ sudo mkdir /usr/lib/apache/
[admin@ids-eng-01 admin]$ sudo mkdir /usr/lib/apache/passwords
```

- Create a password file and add the admin user to it

```
[admin@ids-eng-01 admin]$ sudo /usr/bin/htpasswd –c
/usr/lib/apache/passwords/passwords admin
New password:
Re-type new password:
Adding password for user admin
[admin@ids-eng-01 admin]$
```

- Change permissions on /var/lib/apache/passwords/passwords so that only the web server can read it

```
[admin@ids-eng-01 admin]$ sudo chown apache:apache
/usr/lib/apache/passwords/passwords
[admin@ids-eng-01 admin]$ sudo chmod 600
/usr/lib/apache/passwords/passwords
```

3.4 Restart the apache web server.

```
[admin@ids-eng-01 admin]$ sudo /etc/init.d/httpd restart
Password:
Stopping httpd:                                [  OK  ]
Starting httpd:                                [  OK  ]
[admin@ids-eng-01 admin]$
```

4. Install and configure SnortCenter.

4.1 Install SnortCenter.

- Decompress SnortCenter from /home/admin/packages/snortcenter-v1.0-RC1.tar.gz to /home/admin/packages/www

```
[admin@ids-eng-01 admin]$ cd /home/admin/packages/
[admin@ids-eng-01 admin]$ tar xvzf snortcenter-v1.0-RC1.tar.gz
```

- Rename and move /home/admin/www to /var/www/html/snortcenter

```
[admin@ids-eng-01 admin]$ sudo mv /home/admin/packages/www
/var/www/html/snortcenter
```

4.2 Connect to mysql and create the snortcenter database.

```
[admin@ids-eng-01 admin]$ mysql -u root -p
Enter password: <password from step 2.3>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6 to server version: 3.23.54

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create database snortcenter;
Query OK, 1 row affected (0.02 sec)

mysql>
```

- Exit mysql

```
mysql> exit
Bye
```

4.3 Configure SnortCenter.

- Edit the SnortCenter configuration file

```
[admin@ids-eng-01 packages]$ sudo vi /var/www/html/snortcenter/config.php
```

- Change: $DBlib_path = "./adodb/";
  To: $DBlib_path = "../adodb";

- Change: $DB_password = "";
  To: $DB_password = "<password from step 2.3>";

- Change: $hidden_key_num = "0";
  To: $hidden_key_num = "<random number, say 5409809434435>";

- Save the file

4.4 Initialize the Snort database.

In a web browser on a laptop or desktop connected to the management network, load the following URL:

http://10.13.10.101/acid

You should be prompted with a window similar to the one in Figure 1-7.

Figure 1-7

Log in as admin, using the password you set in step 3.3. Once you're logged in, you'll be presented with a page that looks like the one in Figure 1-8.

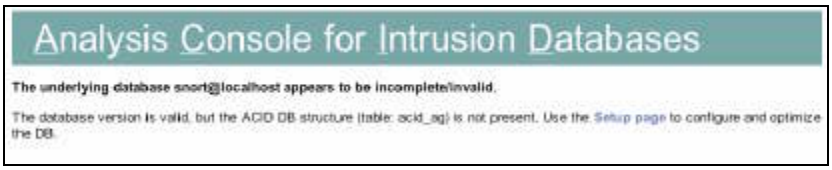


Figure 1-8

Click the "Setup Page" link to create the database structure. On the following page (DB Setup page), click the button labeled "Create ACID AG" (see Figure 1-9). You'll see a message saying that 4 tables were successfully created, as well as a list of operations with a status of "DONE".

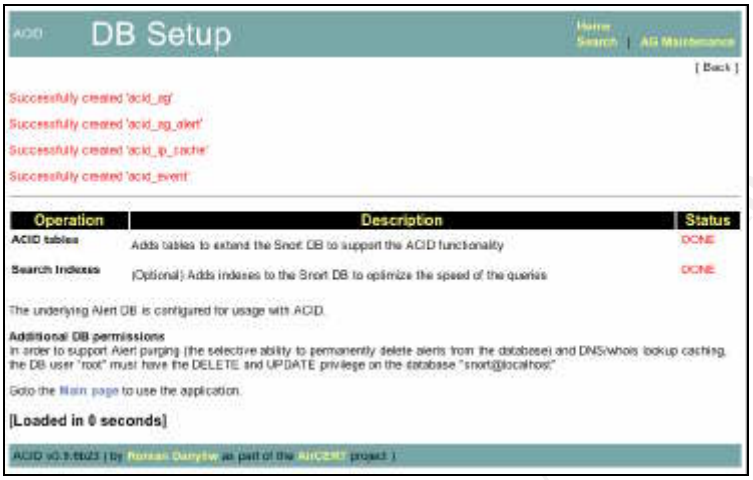


Figure 1-9

Click the "Main Page" link to load the main ACID console page, shown in Figure 1-10. The Snort database has now been updated, and is ready to start receiving data.

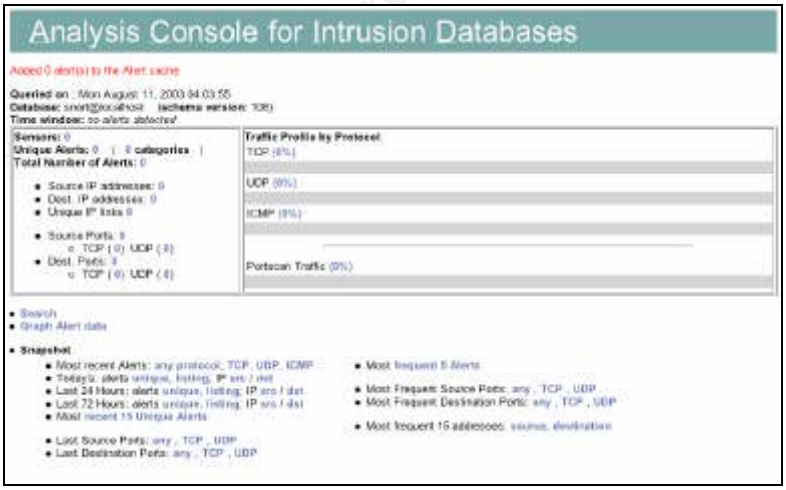


Figure 1-10

4.5 Initialize the SnortCenter database.

In a web browser on a workstation connected to the management network, load the following URL:

http://10.13.10.101/snortcenter

You should see a window similar to the one in Figure 1-11.

Figure 1-11

4.6 Change admin password.

Click the “Admin” link, and login as admin, with “change” as the password. This will load the main SnortCenter page. From the main SnortCenter page, click “Admin, User Administration, View Users”

- click the small edit icon (looks like a piece of paper) next to the **admin** entry

- Edit the password for **admin** (keep this in a safe place)

- Edit the email address

- Click “Update”

At this point, our initial engine installation is complete. The next major step is to install the sensors. Once the sensors are configured, we’ll come back to SnortCenter and ACID.

Sensor Installation

Since both of our sensors are identical except for a few subtle differences, the following section only details the configuration of one sensor, the public sensor. Make the appropriate adjustments when configuring the private sensor.

5. Basic sensor preparation.*

5.1 Turn off unnecessary services.

- Logon as root

```
[root@ids-public-01 pkg]#  chkconfig --list | grep :on
                kudzu           0:off   1:off   2:off   3:on    4:on    5:on    6:off
                syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
                netfs           0:off   1:off   2:off   3:on    4:on    5:on    6:off
                network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
                random          0:off   1:off   2:on    3:on    4:on    5:on    6:off
                rawdevices      0:off   1:off   2:off   3:on    4:on    5:on    6:off
                keytable        0:off   1:on    2:on    3:on    4:on    5:on    6:off
                apmd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
                atd             0:off   1:off   2:off   3:on    4:on    5:on    6:off
                gpm             0:off   1:off   2:on    3:on    4:on    5:on    6:off
                iptables        0:off   1:off   2:on    3:on    4:on    5:on    6:off
                sshd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
                sendmail        0:off   1:off   2:on    3:on    4:on    5:on    6:off
                rhnsd           0:off   1:off   2:off   3:on    4:on    5:on    6:off
                crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
                anacron         0:off   1:off   2:on    3:on    4:on    5:on    6:off
[root@ids-public-01 /]# chkconfig --level 0123456 keytable off
[root@ids-public-01 /]# chkconfig --level 0123456 kudzu off
[root@ids-public-01 /]# chkconfig --level 0123456 sendmail off
[root@ids-public-01 /]# chkconfig --level 0123456 netfs off
```

5.2 Add a non-root user so we're not logging into the box as root.

```
[root@ids-public-01 /]# useradd admin
[root@ids-public-01 /]# passwd admin
Changing password for user admin.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

5.3 Add the non-root user to the sudoers list.

```
[root@ids-public-01 /]# vi /etc/sudoers
```

* Section 5 should not be considered a complete Red Hat hardening guide. Additional resources should be used to further secure the operating system.

- Add: admin ALL=(ALL) ALL

- Save the file

5.4 Log off of root account, log back on as admin.

5.5 Turn off sshv1 and prevent root logon over ssh. If this is the first time the sudo command has been invoked, a quick lecture will be displayed, and you will be prompted for the admin account password.

```
[admin@ids-public-01 admin]$ sudo vi /etc/ssh/sshd_config
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

        #1) Respect the privacy of others.
        #2) Think before you type.
        #3) With great power comes great responsibility.

Password: <enter admin password> (keep this in a safe place)
```

- Change: #Protocol 1,2
  To: Protocol 2

- Change: #PermitRootLogin yes
  To: PermitRootLogin no

- Save the file

5.6 Replace /etc/motd with the text from Table 1-6, or something similar: (include leading and trailing blank lines)

Table 1-6

```
**************************************************************************
        NOTICE TO USERS

        This computer system is for authorized use only. Users (authorized or
        unauthorized) have no explicit or implicit expectation of privacy.

        Any or all uses of this system and all files on this system may be
        intercepted, monitored, recorded, copied, audited, inspected, and
        disclosed to authorized site and law enforcement personnel.

        By using this system, the user consents to such interception, monitoring,
        recording, copying, auditing, inspection, and disclosure at the
        discretion of authorized site.
```

```
        Unauthorized or improper use of this system may result in administrative
        disciplinary action and civil and criminal penalties. By continuing to use
        this system you indicate your awareness of and consent to these terms and
        conditions of use. LOG OFF IMMEDIATELY if you do not agree to the
        conditions stated in this warning.

*************************************************************************
```

5.7 Create /etc/hosts entries for the engine and sensors.

Since we may or may not be using DNS servers, we'll add host entries for our system so that names appear in log files, etc., rather than IP addresses.

```
[admin@ids-public-01 admin]$ sudo vi /etc/hosts
```

- Edit the third line to read:

```
127.0.0.1           localhost.localdomain localhost
```

- Add the following lines:

```
192.168.100.10      ids-eng-01
192.168.100.11      ids-public-01
192.168.100.12       ids-private-01
```

5.8 Remove eth1's IP Address.

Previously in the guide we talked about the hole created by bridging the public and private networks with the management network. We'll remove eth1's IP address, but keep the interface active. This way, no host on the public network will be aware of the interface, but it will still be able to sniff traffic.

```
[admin@ids-public-01 admin]$ sudo vi /etc/sysconfig/network-scripts/ifcfg-eth1
```

- ensure that the file contains ONLY the lines from Table 1-7

Table 1-7

```
DEVICE=eth1
BOOTPROTO=static
ONBOOT=yes
```

- Save the file

5.9 Restart eth1.

```
[admin@ids-public-01 admin]$ sudo /sbin/ifdown eth1
```

```
[admin@ids-public-01 admin]$ sudo /sbin/ifup eth1
```

5.10 Create a folder in admin's home (/home/admin) folder called "packages", and copy the following files from your removable media into that folder:

snort-2.0.1.tar.gz
snortcenter-agent-v1.0-RC1.tar.gz
perl-Net-SSLeay-1.23-0.dag.rh90.i386.rpm

5.11 Create the /var/log/snort folder for Snort to save its logs in.

```
[admin@ids-public-01 admin]$ sudo mkdir /var/log/snort
```

5.12 Reboot the server for our changes to take affect.

6. Compile Snort, Install NetSSLeay and Configure SnortCenter Agent

6.1 Compile Snort.

```
[admin@ids-public-01 admin]$ cd /home/admin/packages/
[admin@ids-public-01 packages]$ tar xzf snort-2.0.1.tar.gz
[admin@ids-public-01 packages]$ cd /home/admin/packages/snort-2.0.1
[admin@ids-public-01 snort-2.0.1]$ ./configure --with-mysql
[admin@ids-public-01 snort-2.0.1]$ make
[admin@ids-public-01 snort-2.0.1]$ sudo make install
[admin@ids-public-01 snort-2.0.1]$ cd /home/admin
```

6.2 Install NetSSLeay.

```
[admin@ids-public-01 admin]$ sudo rpm -ivh /home/admin/packages/perl-Net-
SSLeay-1.23-0.dag.rh90.i386.rpm
Preparing...          ########################################### [100%]
   1:perl-Net-SSLeay ########################################### [100%]
```

6.3 Configure SnortCenter Agent.

```
[admin@ids-public-01 admin]$ cd /home/admin/packages/
[admin@ids-public-01 packages]$ tar xzf snortcenter-agent-v1.0-RC1.tar.gz
```

- Move /home/admin/packages/sensor to /home/admin

```
[admin@ids-public-01 packages]$ mv /home/admin/packages/sensor
/home/admin/
```

- Run the SnortCenter setup script

```
[admin@ids-public-01  packages$ cd /home/admin/sensor
[admin@ids-public-01 sensor]$ sudo ./setup.sh

*******************************************************************************
*       Welcome to the SnortCenter Sensor Agent setup script, version 1.0 RC1
*
*******************************************************************************

Installing Sensor in /home/admin/sensor ...

*******************************************************************************
The Sensor Agent uses separate directories for configuration files and log files.
Unless you want to place them in a other directory, you can just accept the
defaults.

Config file directory [/home/admin/sensor/conf]: <ENTER>
Log file directory [/home/admin/sensor/log]: <ENTER>

*******************************************************************************
SnortCenter Sensor Agent is written entirely in Perl. Please enter the full path to
the
Perl 5 interpreter on your system.

Full path to perl (default /usr/bin/perl): <ENTER>

Testing Perl ...
Perl seems to be installed ok

*******************************************************************************
SnortCenter Sensor Agent needs Snort to be installed, 'As if you didn't know :-)'
Please enter the full path to snort binary.

Full path to snort (default /usr/local/bin/):  <ENTER>

Ok, found Snort Version 2.0.1 (Build 88)

Snort Rule config file directory [/home/admin/sensor/rules/]:  <ENTER>

*******************************************************************
Operating system name:    Redhat Linux
Operating system version: 9.0

*******************************************************************
SnortCenter Sensor Agent uses its own password protected web server
The setup script needs to know :
 - What port to run the Sensor Agent on. There must not be another
```

```
   service already using this port.
 - What ip address to listen on.
 - The login name required to access the Sensor Agent.
 - The password required to access the Sensor Agent.
 - The hostname of this system that the Sensor Agent should use.
 - If the Sensor Agent should use SSL (if your system supports it).
 - Whether to use ip access control.
 - Whether to start Snortcenter Sensor Agent at boot time.

Sensor port (default 2525): <ENTER>

If this host has multiple IP addresses,
the server can be configured to listen on
only one address (default any): 192.168.100.11
Login name (default admin): <ENTER>
Login password:
Password again:  (keep this in a safe place)
Sensor host name (default ids-public-01): <ENTER>
Use SSL (y/n): y
**********************************************************************************************
*
The Sensor Agent can be configured allow access only from certain IP
addresses.
Hostnames (like foo.bar.com) and IP networks (like 10.254.3.0 or
10.254.1.0/255.255.255.128)
can also be entered.
You should limit access to your sensor to trusted addresses like the
SnortCenter Management Console, especially if it is accessible from the Internet.
Otherwise, anyone who guesses your password will have complete control of
your system.
You can enter multiple addresses by typing a space between them like
(127.0.0.1 foo.bar.com)

Allowed IP addresses (default localhost):192.168.100.10
Start Sensor at boot time (y/n): y
**********************************************************************
Creating Sensor Agent config files..
..done

Inserting path to perl into scripts..
..done
Creating start and stop scripts..
..done
Copying config files..
..done
Configuring SnortCenter Sensor Agent to start at boot time..
Created init script /etc/rc.d/init.d/sensor
```

```
..done
Creating uninstall script /home/admin/sensor/conf/uninstall.sh ..
..done
Changing ownership and permissions ..
..done
Attempting to start Sensor Agent..
Starting SnortCenter Sensor Agent server in /home/admin/sensor
..done
*******************************************************************************
SnortCenter Sensor Agent has been installed and started successfully.
You can now create and configure the sensor in the SnortCenter Management
Console.
Or use your webbrowser to go to

  https://ids-public-01:2525/

and login with the name and password you entered previously.

Because the Sensor Agent uses SSL for encryption only, the certificate
it uses is not signed by one of the recognized CAs such as Verisign.
When you first connect to the Sensor Agent, your browser will ask you if
you want to accept the certificate presented, as it does not recognize the CA.
Say yes.
```

The public sensor installation is now complete. If you haven't already, repeat sections 5 and 6 for the private sensor, making the proper substitutions. Once both sensors are complete, proceed to section 7 of the guide.

Configuration

7. Add and configure sensors in SnortCenter.

7.1 Add sensors to SnortCenter.

In a web browser on a workstation connected to the management network, load the following URL:

http://10.13.10.101/snortcenter

- Login as admin

- Click "Sensor Console, Add Sensor"

Using ids-public-01 as an example, enter the information in Table 1-8 to create a new sensor:

Table 1-8

Sensor Name: ids-public-01 **Sensor IP:** 192.168.100.11 **Port #:** 2525 **Sensor Username:** admin (from SnortCenter agent installation, section 6.3) **Sensor Password:** (from SnortCenter agent installation, section 6.3) **Sensor Agent Type:** SnortCenter Agent v.1 (SSL enabled) **Interface to sniff:** eth1 **Snort command line:** -o

- click “Save” to add the new sensor

At the main SnortCenter page, you should see that ids-public-01 (shortened to idspublic01) has been added to the list of sensors and has a yellow background. In addition, you will notice that Snort is not (yet) running on ids-public-01, as seen in Figure 1-12.

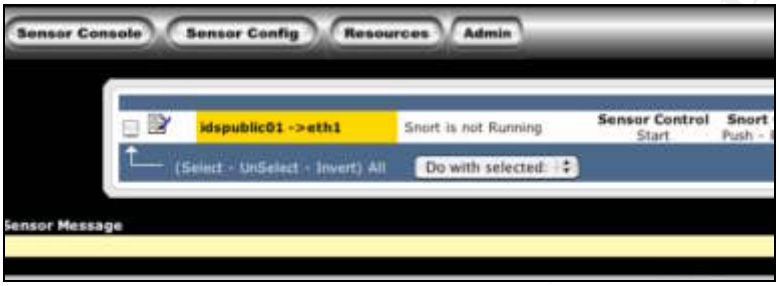


Figure 1-12

7.2 Configure Snort Rules.*

Snort tests each packet that it sniffs from the network against a list of activated rules. We’ll use SnortCenter to update the list of available rules, as well as select and customize what rules we want to watch for.

7.3 Update Snort Rules from the Internet.

- Click “Admin, Import/Update Rules, Update from Internet”

You should see a page titled “Update Report”, listing new variables, etc.

7.4 Create a rule template.

More information about Snort rules can be found in chapter 2 of the Snort Users Manual, located at http://www.snort.org/docs/writing_rules/index.html

Templates allow us to add or remove a sensor without losing the rule settings. We'll create templates for each network, then apply the templates to the appropriate sensor.

- Click "Resources, Create Rule Template"

Using ids-public-01 as an example, enter the information in Table 1-9 to create a new template:

Table 1-9

Template Name: IDS-Public-01 **Template Description:** Public Network

- Click "Save" to create the template

The next page is where we'll apply templates to the sensors, then enable rules within the template.

- Click the red "X" next to "IDS-Public-01" (The red "X" should change to a green check mark)

- Click the small edit icon (looks like a piece of paper) next to the template named "IDS-Public-01"

To start, we'll enable all rules from this page (see Figure 1-13). Rules are categorized into scopes, which can be displayed via the "Category Scope" drop-down box. You should see categories like "attack-response.rules, backdoor.rules", etc. Select "All rules" from the list.

- At the bottom of the page, click "Do with selected, Activate Category Scope"

All of the rules in each category scope should now have a green push pin and check mark icon next to them. For now, we'll leave all of the rules enabled. Later in the guide we'll disable some rules as we tune our sensors.

7.4 Enable Snort Variables.

Snort rules come with many common variables, which can be changed to better accommodate our sensors. For example, we might consider changing the value of the HOME_NET variable from "any" to 10.13.0.0/16 for our private sensor. We'd do this because some Snort rules specify that a packet must be moving in a certain direction to be a positive match. Let's say a rule states that a packet must be moving toward the HOME_NET in order to be a match. If we have the HOME_NET variable set to 10.13.0.0/16, then only those packets destined for the 10.13.0.0/16 subnet will trigger the rule. In an ideal environment, setting the HOME_NET variable this way would be a good way to monitor the private lan.

However, if a malicious packet with a different destination address traverses the network, the rule won't be triggered. With this in mind, it's often a good idea to start with the HOME_NET variable set to "any". While there may be a higher occurrence of false positives, a higher level of monitoring can be achieved.

- Click "Sensor Config, Variable Selection"

- Select the desired sensor from the "Sensor Scope" drop-down box

- At the bottom of the page, click "Select", then click "Do with selected, Activate"

7.5 Output Plugins.

Snort can output its alerts in a variety of ways. In our case, we need to configure it to output to a database.

- Click "Resources, Output Plugins, Create Output Plugin"

- Select "Database (Log to a variety of databases)" from the "Select Output Plugin to create" drop-down box

- Click "Select"

Using ids-public-01 as an example, enter the information in Table 1-10 to create a new output plugin:

Table 1-10

Sensor Name: ids-public-01 **DB Name:** snort **DB Type:** mysql **DB Host**: 192.168.100.10 **DB Port:** <blank> **User:** snort **Password:** <enter password from step 2.4> **Ruletype:** log **Encoding:** <blank> **Detail:** <blank>

- Click "Save"

Apply the output plugin to the sensor

- Click "Sensor Config, Output Plugin Selection"

- Select the desired sensor from the "Sensor Scope" drop-down box

- Click the empty check box to the left of the desired output plugin

- Click “Do with selected, Activate”

7.6 Preprocessors*

Preprocessors perform certain advanced functions on packets before they’re handed to Snort’s detection engine. Since we want our sensors to start off as sensitive as possible, we’ll enable all preprocessors.

- Click “Sensor Config, Preprocessor Selection”

- Select the desired sensor from the “Sensor Scope” drop-down box

- At the bottom of the page, click “Select”, then click “Do with selected, Activate”

7.7 Classifications and Reference.

Snort rules contain areas for classification and reference. This extends the reporting functionality and ability to abstract more from rules that contain information in these areas.

- Click “Sensor Config, Classification Selection”

- Select the desired sensor from the “Sensor Scope” drop-down box

- At the bottom of the page, click “Select”, then click “Do with selected, Activate”

- Click “Sensor Config, Reference Selection”

- Select the desired sensor from the “Sensor Scope” drop-down box

- At the bottom of the page, click “Select”, then click “Do with selected, Activate”

7.8 Start the sensor.

Now that we’ve defined the settings for our public sensor, we need to update its configuration in order to begin sniffing traffic.

---

* More information about Snort preprocessors can be found in chapter 2 of the Snort Users Manual, located at http://www.snort.org/docs/writing_rules/index.html

- Click “Sensor Console”

This console page is where the sensors will be listed. In addition, from this page we can stop, start, reload and push a configuration to any and all sensors.

- At the bottom of the page, click “Select”, then click “Do with selected:, Push” (Alternatively, click “Push” under “Snort Configuration File”)

The “Sensor Message” section of the page details the success or failure of the most recent operation. The push operation should have produced a success.

- At the bottom of the page, click “Select”, then click “Do with selected:, Start” (Alternatively, click “Start” under “Snort Configuration File”)

The “Sensor Message” section will most likely contain an error, with details at the bottom of the page. The last two lines of the error should look like Figure 1-13.

Figure 1-13

If we were to look at /home/admin/sensor/rules/snort.eth1.conf on the sensor, we’d find that a Snort rule (sid 1882) has a typo in it. For now, the easiest thing to do is disable this rule.

- Click “Sensor Config, Rule Selection”

- Click the small edit icon (looks like a piece of paper) next to the template named “IDS-Public-01”

- In the upper right-hand corner, enter 1882 in the empty box next to “Find”, and hit ENTER

The rule with sid 1882 should be highlighted in yellow.

- Click the green check mark the the left of the rule

Repeat the steps in this section to push and restart the public sensor. If the push and restart are successful, the sensor will have a green background.

We’ve successfully added the private sensor to SnortCenter. If you haven’t already, repeat section 7 for the private sensor. Once both sensors have been

added and configured, the sensors should both be green in the SnortCenter console. (see Figure 1-14)

Figure 1-14

Monitoring

8. Working with the ACID Console.

8.1 Create an alert.

In a web browser on a workstation connected to the management network, load the following URL:

http://10.13.10.101/acid

Log in as admin, using the password you set in step 3.3. You should see a page similar to the one in Figure 1-15

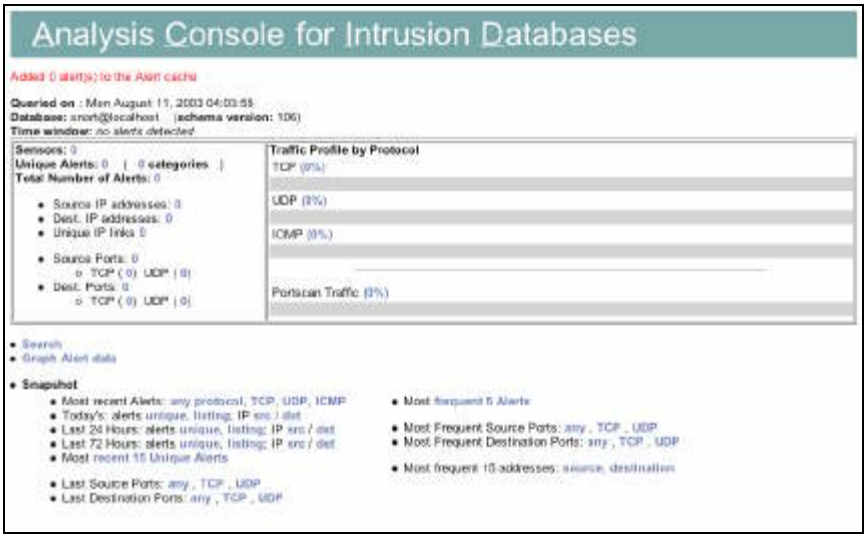


Figure 1-15

As ACID reports, no alerts have been added to the cache. In addition, ACID reports no sensors. We can add an alert, and at the same time verify that our sensors are properly sniffing traffic.

From a workstation on the public segment, send a large ping to a host (10.10.10.144 in the example) on that segment.

```
workstation:~] % ping -c 1 -s 5550 10.10.10.144
```

This should trigger some ICMP Snort rules (sids 368 and 499). If you refresh the main ACID page, you'll notice that 2 alerts have been added, and 1 sensor is now reported.

8.2 View Details of an Alert

- Click the "2" next to "Total Number of Alerts"

ACID lets us view the details of every alert. We can view more details of the two suspected packets. (see Figure 1-16)

Figure 1-16

Drill down further to view the contents of the packets.

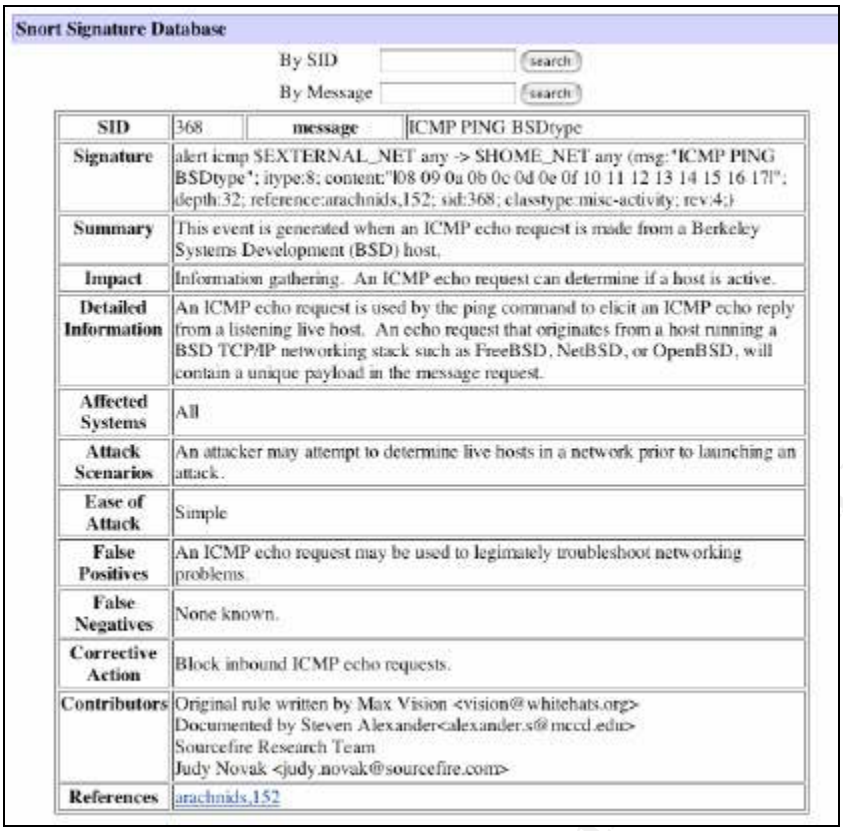
- Click on the ID# of the first alert in the list

The following page shows the details of the packet. Below the Meta and IP sections we see that the packet is an ICMP layer 4 packet, with the packet's contents displayed in the Payload section. Take notice that the length (5550) of the payload is the size of the ping we sent in section 8.1.

Check the snort.org database entry for more information on this alert.

- Click on "snort" under "Triggered Signature"

The following page (which may open in a separate window) details the signature of the alert, as well as other many important bits of information. (see Figure 1-17)



Snort Signature Database

By SID [search]

By Message [search]

SID	368	**message**	ICMP PING BSDtype
Signature	alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP PING BSDtype"; itype:8; content:"|08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17|"; depth:32; reference:arachnids,152; sid:368; classtype:misc-activity; rev:4;)		
Summary	This event is generated when an ICMP echo request is made from a Berkeley Systems Development (BSD) host.		
Impact	Information gathering. An ICMP echo request can determine if a host is active.		
Detailed Information	An ICMP echo request is used by the ping command to elicit an ICMP echo reply from a listening live host. An echo request that originates from a host running a BSD TCP/IP networking stack such as FreeBSD, NetBSD, or OpenBSD, will contain a unique payload in the message request.		
Affected Systems	All		
Attack Scenarios	An attacker may attempt to determine live hosts in a network prior to launching an attack.		
Ease of Attack	Simple		
False Positives	An ICMP echo request may be used to legimately troubleshoot networking problems.		
False Negatives	None known.		
Corrective Action	Block inbound ICMP echo requests.		
Contributors	Original rule written by Max Vision <vision@whitehats.org> Documented by Steven Alexander<alexander.s@mccd.edu> Sourcefire Research Team Judy Novak <judy.novak@sourcefire.com>		
References	arachnids,152		

Figure 1-17

8.3 Deleting Alerts

On some occasions, especially during the first few days of a new sensor's implementation, it's desirable to delete alerts from the database. Improperly configured network devices can often generate false positives, and there's no need to keep these alerts in the database. ACID gives you the ability to delete selected alerts, all alerts on the screen, or even the entire query. To delete the alert we looked at in section 8.2, repeat the steps in that section, but instead of clicking on the alert's ID#, do the following:

- Click the check box next the the ID# of the first alert in the list

- At the bottom of the page, select "Delete" from the "action" drop-down box

- Click "Selected"

On the following page you'll see that there was one "Successful DELETE".

8.4 Archiving Alerts

The process of archiving alerts is similar to deleting them. However, archiving requires that an archive database be available. Appendix IV details the steps required to create an archive database.

Recommended Additions

This section describes some optional packages you may want to add to your distributed NIDS to increase its security, accuracy, and stability.

IP Tables*
If you used the Appendices in this guide to build your distributed NIDS on Red Hat 9, you should have installed iptables on your engine and sensors. IP Tables is a very reliable and easily configured host-based firewall. It's recommended that iptables be configured on your engine to only allow the necessary IP traffic to and from each server.

NTP
Network Time Protocol is a good service to have running on your servers, assuming a host or network-based firewall is installed. NTP allows servers to keep time based on synchronization with other servers. In respect to our distributed NIDS, the engine should sync to two or more public NTP servers, and the sensors should sync to the engine. If you decided to keep the engine isolated from the Internet, the sensors can still sync with the engine so that all NIDS servers are on the same time.

Updates
Updates and patches are critical this day in age. While our NIDS servers are most likely protected by firewalls, it's crucial that they stay up to date, especially Snort and Snort rules within SnortCenter. If you used the Appendices in this guide to build your distributed NIDS on Red Hat 9, you should have installed up2date, a Red Hat Network tool to update your systems. This tool connects to a central server which grabs GPG-signed updates for the OS. If you followed the guide's recommendations and opted not to connect your sensors to the Internet, updates and patches for your Red Hat 9 sensors can be found at the following URL;

https://rhn.redhat.com/errata/rh9-errata.html

Summary

Now that you have a working distributed NIDS, it's probably a good time to spend some time experimenting with SnortCenter, ACID, and their many features. This

* http://www.netfilter.org

guide stops short of guiding the reader through every feature and detail of these tools, so that they can be learned naturally. There are many online resources for Snort, ACID, MySQL and the other packages if problems arise.

While there are many IDS solutions on the market today, most are extremely expensive to implement and maintain. Hopefully this guide will help System and Network Administrators get started down the road to serious intrusion detection.

References

Northcut, Stephen. Network Intrusion Detection An Analyst's Handbook. Indianapolis: New Riders, September 2000

Innella, Paul. The Evolution of Intrusion Detection Systems. 16 November 2001. URL: http://www.securityfocus.com/infocus/1514

Roesch, Martin. Snort Users Manual. 2003. URL: http://www.snort.org/docs/writing_rules

Scott, Stephen J. Snort Installation Manual. August 2002. URL: http://www.snort.org/docs/snort-rh7-mysql-ACID-1-5.pdf

Neohapsis Archives. Current. July 2003. URL: http://archives.neohapsis.com/archives/snort

Moore, Sandra A. Red Hat Linux x86 Installation Guide. 2003. July, 2003. URL: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/install-guide

Laing, Brian. HOWTO GUIDE-Implementing a Network Based Intrusion Detection System. 2000. July 2003. URL: http://www.snort.org/docs/iss-placement.pdf

Danyliw, Roman. ACID: Installation and Configuration. 9 October 2002. July 2003. URL: http://www.andrew.cmu.edu/~rdanyliw/snort/acid_config.html

Appendix I

Engine OS Installation

Since the engine is center of our NIDS, we'll build it first. Insert RedHat 9 CD 1 into the CD-ROM drive of the engine and boot the machine up. You may need to configure the machine to boot from the CD-ROM drive.
Once the machine boots, the installer should take you to a text-based screen with a "redhat" logo at the top, and a "boot:" prompt at the bottom. Follow the instructions to "install or upgrade Red Hat Linux in graphical mode". Optionally, a text mode is available by typing "linux text" at the boot prompt, followed by hitting the <Enter> key. Both options will take you through the same steps, though this guide will proceed with the graphical installation.
The following screen will ask if you want to test the CD media before installation. Choose "Skip" for now, as this step can take a while. If you're not confident with your CD media, go ahead and run this test.
The next screen begins the graphical part of the installation, shown in the figures below.

Figure 2-1

-Welcome to Red Hat Linux (figure 2-1)

- Click "Next"

Figure 2-2

- Language Selection (figure 2-2)
    - Click “English” (or your preferred language)
    - Click “Next”

Figure 2-3

- Keyboard Selection (figure 2-3)
    - Click “U.S. English”
    - Click “Next”

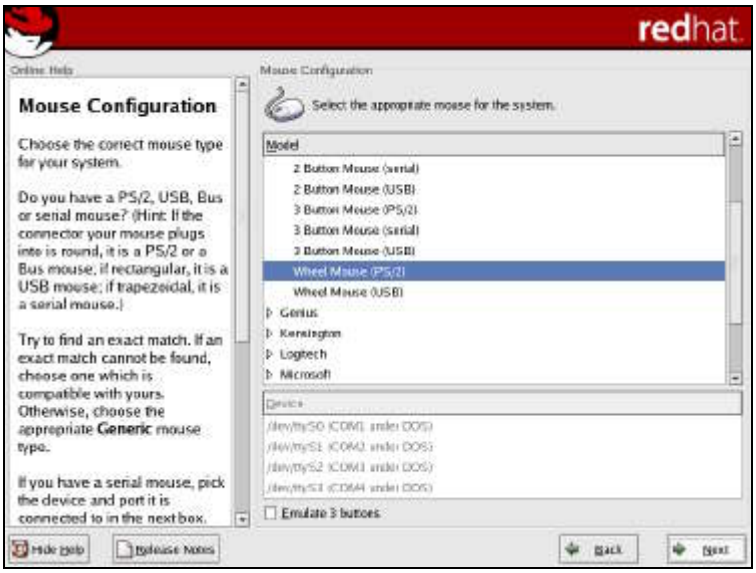


Figure 2-4

- Mouse Configuration (figure 2-4)
  - Select your mouse type
  - Click “Next”

Figure 2-5

- Installation Type (figure 2-5)
  - Click “Custom”
  - Click “Next”

Figure 2-6

- Disk Partitioning Setup (figure 2-6)
  - Select “Manually Partition”
  - Click “Next”

Figure 2-7

- Disk Setup (figure 2-7)

  For our engine, we will create a large partition separate from the OS to house our database. Since it may eventually outgrow the current disk, we should keep it in a place that will be easy to move later.

  Use the following recommendations for your partitioning layout. If you have reasons to change the size of any parts of the disk, feel free to do so. Click “New” to create the following partitions:

Mount Point	File System Type	Size (MB)
/boot	ext3	100
/	ext3	1024
Not Applicable	swap	1024
/usr	ext3	4096
/var	ext3	4096
/db	ext3	Fill to maximum allowable size

Your final partition layout should look similar to the example in figure 2-8.

- Click “Next”

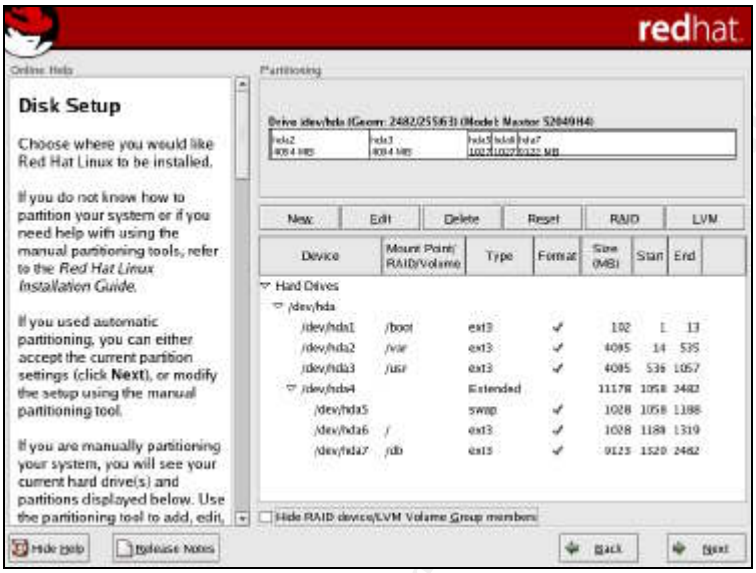


Figure 2-8

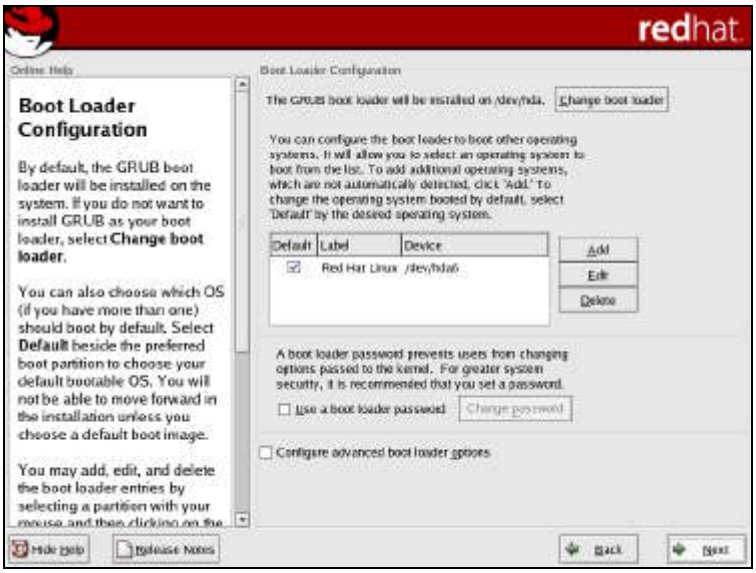


Figure 2-9

- Boot Loader Configuration (Figure 2-9)
  - Ensure everything looks OK, and optionally set a password for the boot loader, or configure advanced options.
  - Click “Next”

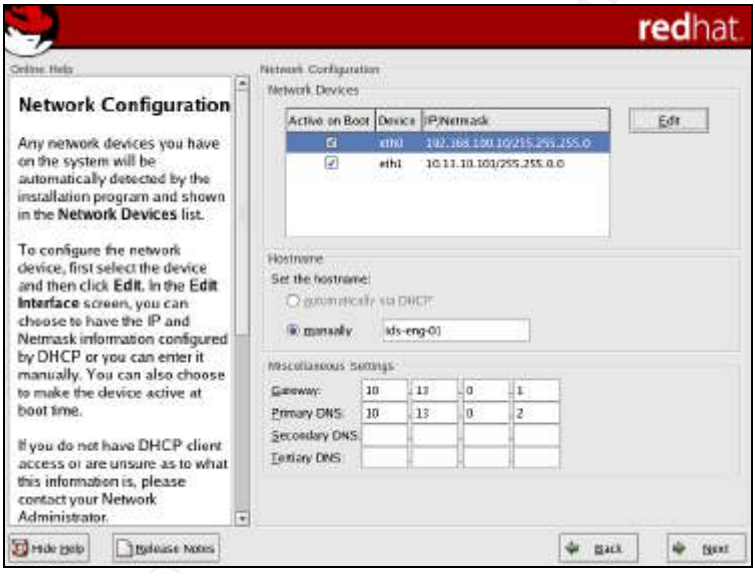


Figure 2-10

- Network Configuration (figure 2-10)
  Two network devices should be shown, eth0 and eth1. Eth0 will be the interface connected to the management network, and eth1 will connect back to the private lan.
  - Highlight eth0 and click “Edit” to configure settings for eth0
    - Deselect “Configure using DHCP”
    - IP Address: 192.168.100.10
    - Subnet Mask: 255.255.255.0

    - Click “OK”
- Highlight eth1 and click “Edit” to configure settings for eth1
    - Deselect “Configure using DHCP”
    - IP Address: 10.13.10.101
    - Subnet Mask: 255.255.0.0
    - Click “OK”
- Set the hostname manually : ids-eng-01
- Gateway : 10.13.0.1
- Primary DNS : 10.13.0.2
- Secondary DNS : blank
- Tertiary DNS : blank
- Click “Next”

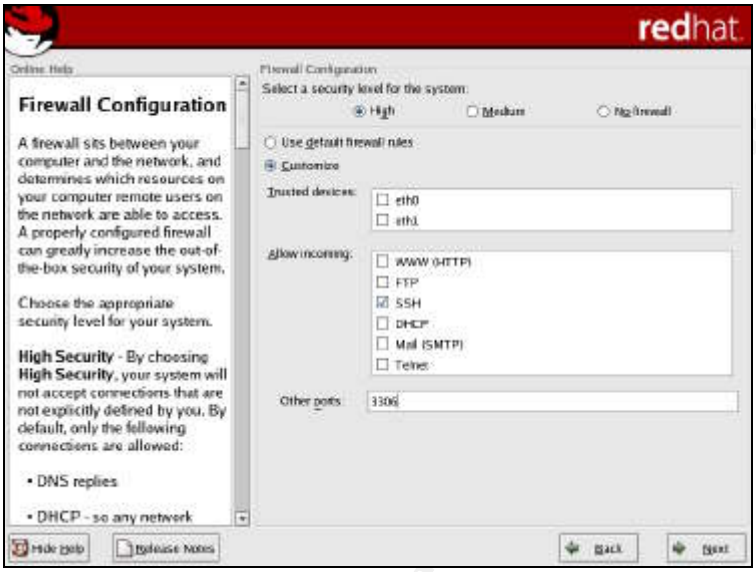


Figure 2-11

- Firewall Configuration (Figure 2-11)
    - Select a security level for the system: High
    - Allow incoming: WWW, SSH
    - Other ports: 3306 (MySQL)
    - Click “Next”

Figure 2-12

- Additional Language Support (figure2-12)
  - Select any additional languages needed
  - Click “Next”

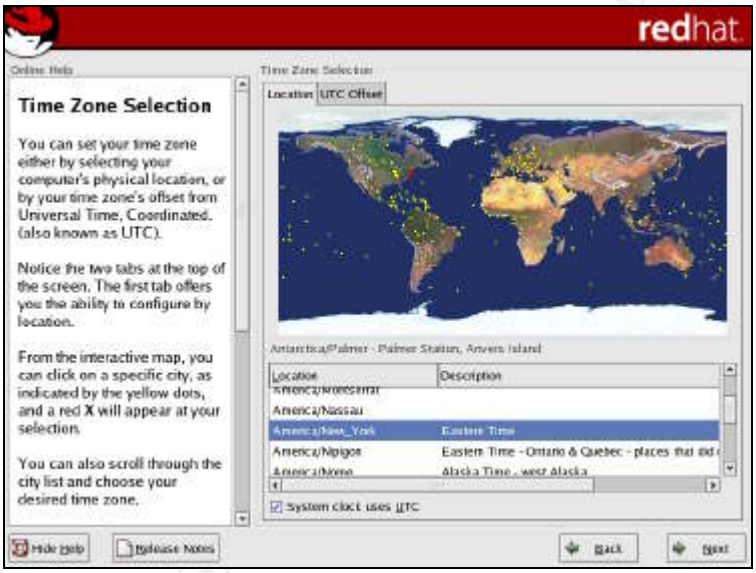


Figure 2-13

- Time Zone Selection (figure 2-13)
  - Select time zone via the map or the drop down list
  - Click “System clock uses UTC”
  - Click the “UTC Offset” tab
    - Select the proper UTC offset for your time zone
    - Click “Use daylight savings time (US Only)” (optional)
  - Click “Next”

Figure 2-14

- Set Root Password (figure 2-14)
    - Enter and confirm your root password. Keep this in a safe place.
    - Click “Next”

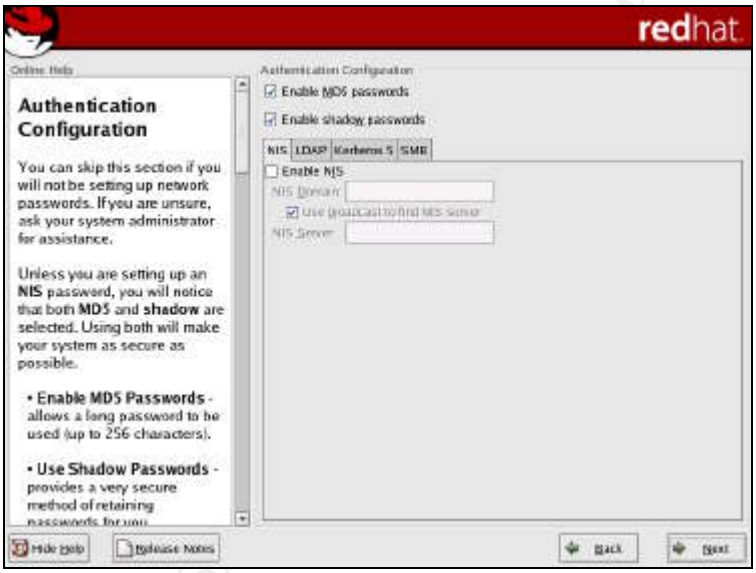


Figure 2-15

- Authentication Configuration Figure 1-20 0018
    - Click “Next”

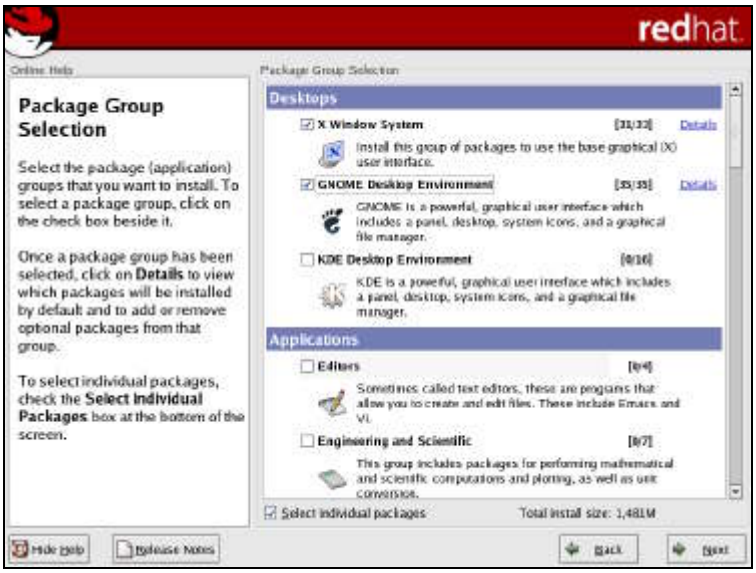


Figure 2-16

- Package Group Selection (figure 2-16)
  - Click “Select individual packages”
  - Click “Next”

Figure 2-17

- Individual Package Selection (figure 2-17)
  Select the packages listed below. Add in any optional packages that aren’t listed.
  - Amusements
    - None
  - Applications
    - bc, bind-utils, curl, diffutils, lokkit, m4, mailx, mysql, mysql-devel, mysql-server, openssh, openssh-clients, rsync, sudo, tcpdump, time, traceroute, unzip, zip

- Development
    - Binutils, bison, compat-gcc, compat-gcc++, compat-libstdc++-devel, cpp, gcc, gcc-c++, glibc-devel, glibc-kernheaders, libcap-devel, libpcap, lsof, make, patch, perl, php, php-mysql, rhnlib
- System Environment
    - Acl, anacron, apmd, at, compat-libstdc++, crontabs, gpm, httpd, iptables, libcap, libstdc++, logrotate, man, mod_perl, mod_ssl, ntp, openssh-server, procmail, sendmail, up2date

Figure 2-18

- Unresolved Dependencies (figure 2-18)
    - Click “Install packages to satisfy dependencies”
    - Click “Next”

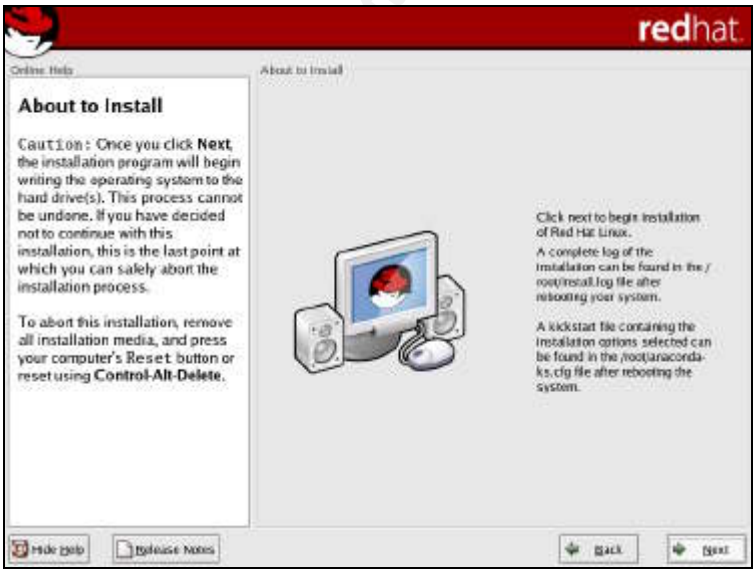


Figure 2-19

- About to Install (figure 2-19)
    - Click “Next”

Figure 2-20

- Boot Diskette Creation (figure 2-20)
    - Click “No, I do not want to create a boot diskette”
    - Click “Next”

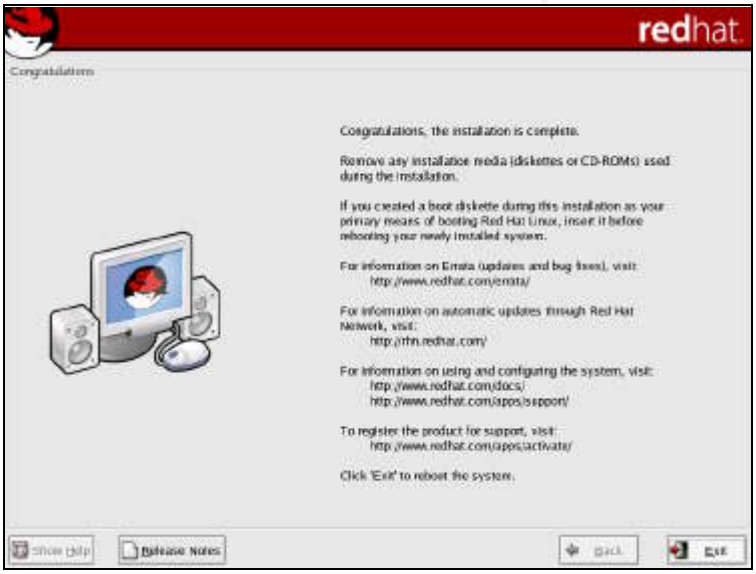


Figure 2-21

- Congratulations (figure 2-21)
    - Click “Exit” to reboot

Appendix II

Sensor OS Installation

Insert RedHat 9 CD 1 into the CD-ROM drive of the engine and boot the machine up. You may need to configure the machine to boot from the CD-ROM drive. Once the machine boots, the installer should take you to a text-based screen with a “redhat” logo at the top, and a “boot:” prompt at the bottom. Follow the instructions to “install or upgrade Red Hat Linux in graphical mode”. Optionally, a text mode is available by typing “linux text” at the boot prompt, followed by hitting the <Enter> key. Both options will take you through the same steps, though this guide will proceed with the graphical installation.
The following screen will ask if you want to test the CD media before installation. Choose “Skip” for now, as this step can take a while. If you’re not confident with your CD media, go ahead and run this test.
The next screen begins the graphical part of the installation, shown in the figures below.

Figure 3-1

-Welcome to Red Hat Linux (Figure 3-1)

- Click “Next”

Figure 3-2

- Language Selection (Figure 3-2)
    - Click “English” (or your preferred language)
    - Click “Next”

Figure 3-3

- Keyboard Selection (Figure 3-3)
    - Click “U.S. English”
    - Click “Next”

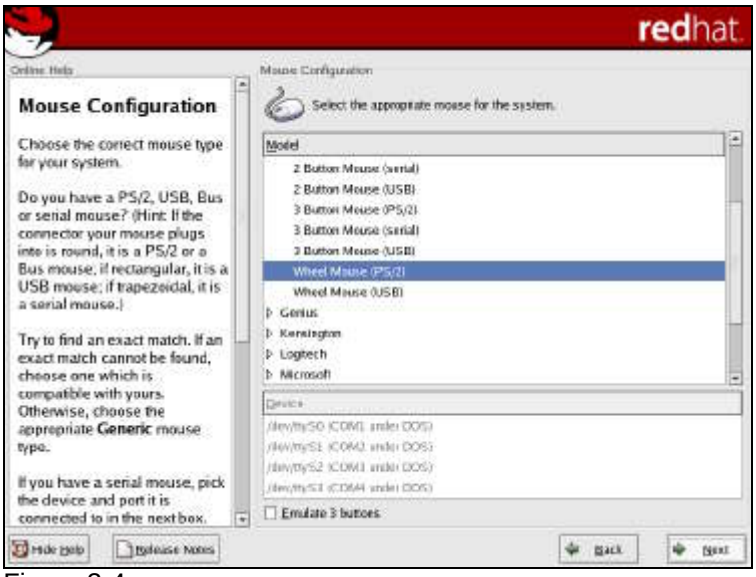


Figure 3-4

- Mouse Configuration (Figure 3-4)
  - Select your mouse type
  - Click “Next”

Figure 3-5

- Installation Type (Figure 3-5)
  - Click “Custom”
  - Click “Next”

Figure 3-6

- Disk Partitioning Setup (Figure 3-6)
  - Select “Manually Partition”
  - Click “Next”

Figure 3-7

- Disk Setup (Figure 3-7)

  Use the following recommendations for your partitioning layout. If you have a reason to change the size of any parts of the disk, feel free to do so. Click “New” to create the following partitions:

Mount Point	File System Type	Size (MB)
/boot	ext3	100
/	ext3	Fill to maximum allowable size
Not Applicable	swap	1024

/usr	ext3	4096
/var	ext3	4096

Your final partition layout should look similar to the example in Figure 3-8.

- Click “Next”

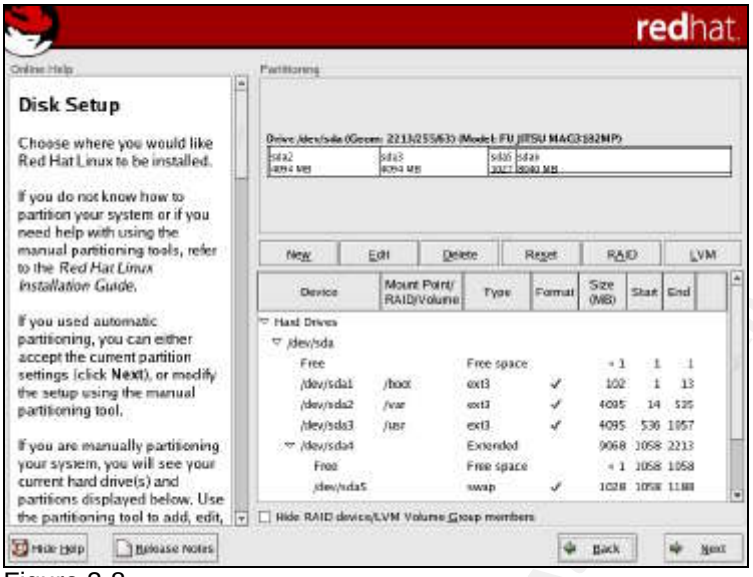


Figure 3-8

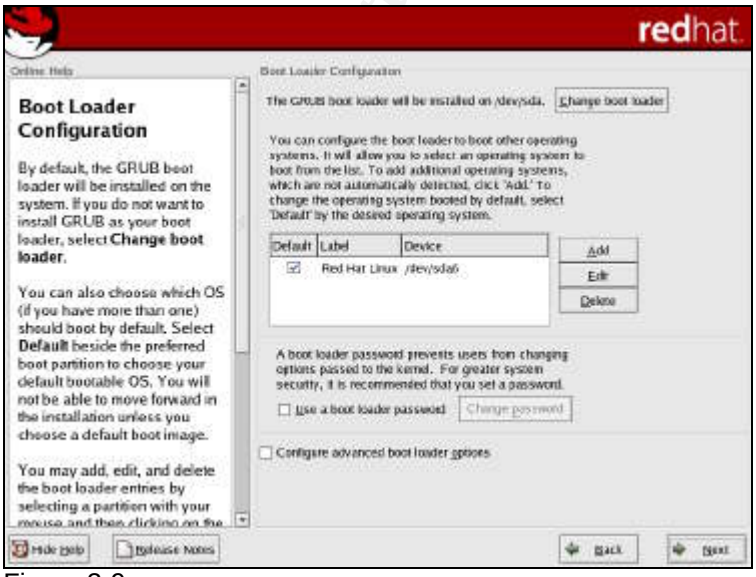


Figure 3-9

- Boot Loader Configuration (Figure 3-9)
    - Ensure everything looks OK, and optionally set a password for the boot loader, or configure advanced options.
    - Click “Next”

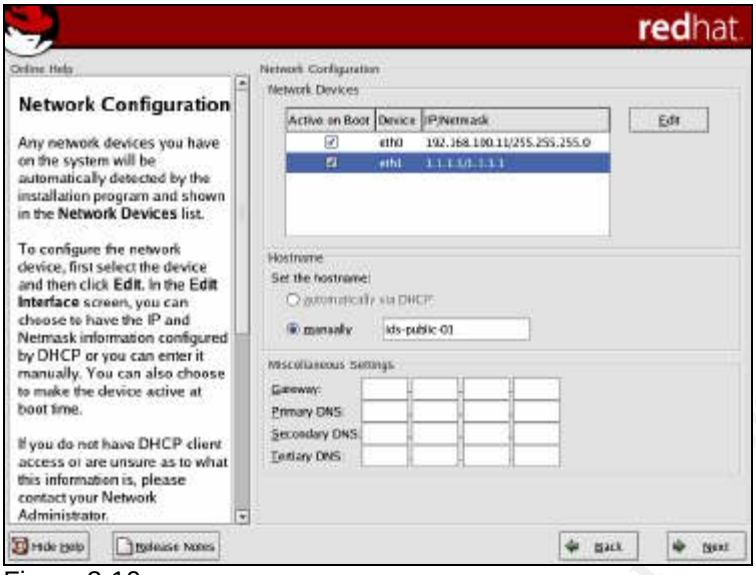


Figure 3-10

- Network Configuration (Figure 3-10)

    Two network devices should be shown, eth0 and eth1. Eth0 will be the interface connected to the management network, and Eth1 will be our sniffing interface. Although we’ll set the IP and netmask for eth1 to 1.1.1.1 and 1.1.1.1 now, we’ll adjust that later.

    - Highlight eth0 and click “Edit” to configure settings for eth0
        - Deselect “Configure using DHCP”
        - IP Address: 192.168.100.11 (public) or 192.168.100.12 (private)
        - Subnet Mask: 255.255.255.0
        - Click “OK”
    - Highlight eth1 and click “Edit” to configure settings for eth1
        - Deselect “Configure using DHCP”
        - IP Address: 1.1.1.1
        - Subnet Mask: 1.1.1.1
        - Click “OK”
    - Set the hostname manually : ids-public-01 or ids-private-01
    - Gateway : blank
    - Primary DNS : blank
    - Secondary DNS : blank
    - Tertiary DNS : blank
    - Click “Next”

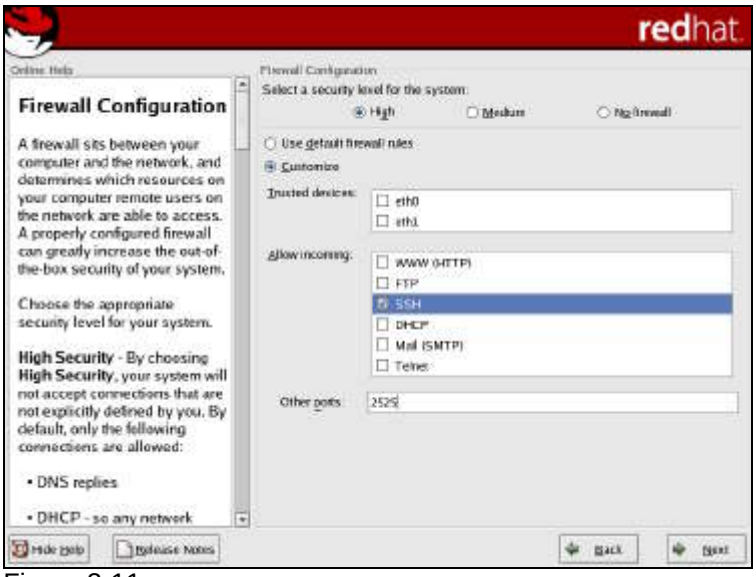


Figure 3-11

- Firewall Configuration (Figure 3-11)
  - Select a security level for the system: High
  - Allow incoming: SSH
  - Other ports: 2525 (Snortcenter)
  - Click “Next”

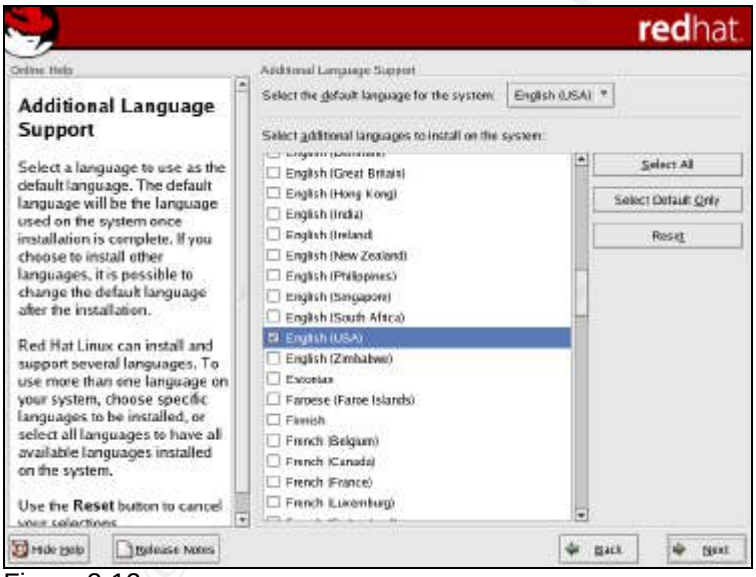


Figure 3-12

- Additional Language Support (figure2-12)
  - Select any additional languages needed
  - Click “Next”

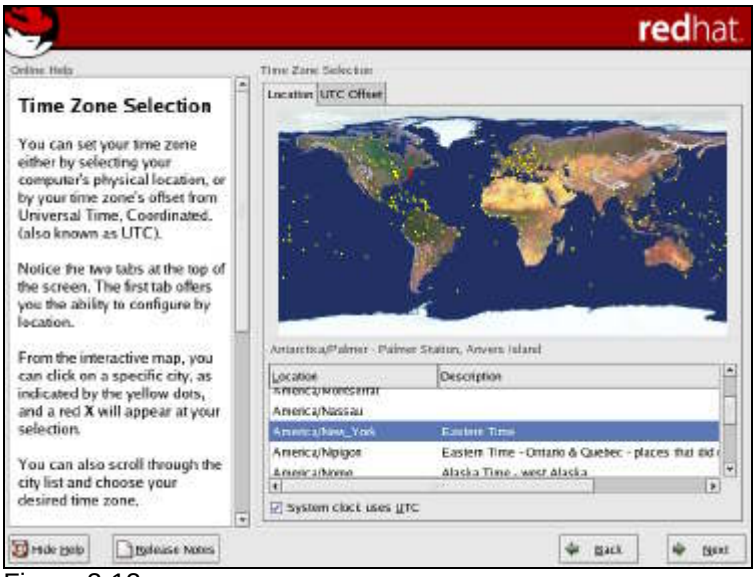


Figure 3-13

- Time Zone Selection (Figure 3-13)
  - Select time zone via the map or the drop down list
  - Click “System clock uses UTC”
  - Click the “UTC Offset” tab
    - Select the proper UTC offset for your time zone
    - Click “Use daylight savings time (US Only)” (optional)
  - Click “Next”

Figure 3-14

- Set Root Password (Figure 3-14)
  - Enter and confirm your root password. Keep this in a safe place.
  - Click “Next”

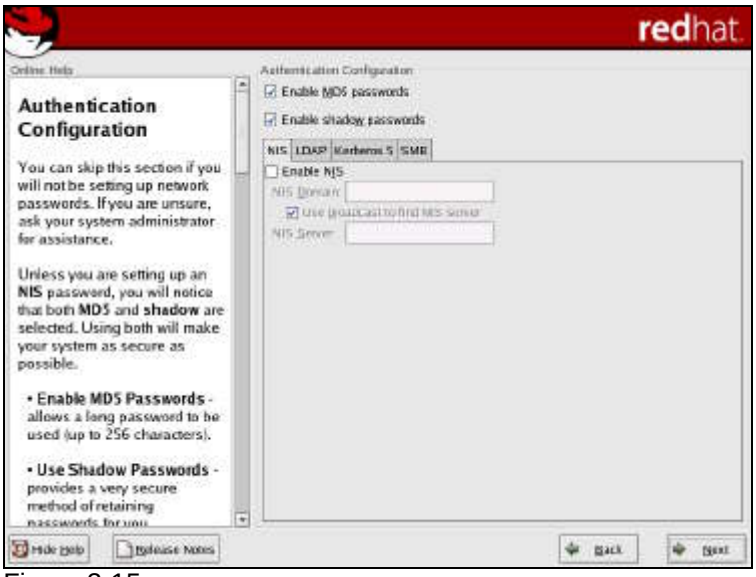


Figure 3-15

- Authentication Configuration (Figure 3-15)
    - Click “Next”

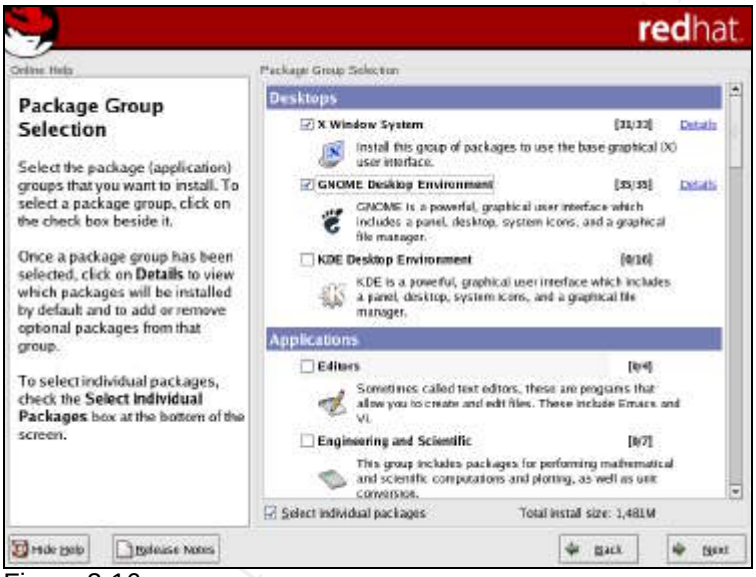


Figure 3-16

- Package Group Selection (Figure 3-16)
    - Click “Select individual packages”
    - Click “Next”

Figure 3-17

- Individual Package Selection (Figure 3-17)
  Select the packages listed below. Add in any optional packages that aren't listed.
  - Amusements
    - None
  - Applications
    - bc, bind-utils, curl, diffutils, lokkit, m4, mailx, mysql, mysql-devel, openssh, openssh-clients, rsync, sudo, tcpdump, time, traceroute, unzip, zip
  - Development
    - Binutils, bison, compat-gcc, compat-gcc++, compat-libstdc++-devel, cpp, gcc, gcc-c++, glibc-devel, glibc-kernheaders, libcap-devel, libpcap, lsof, make, patch, perl, rhnlib
  - System Environment
    - Acl, anacron, apmd, at, compat-libstdc++, crontabs, gpm, iptables, libcap, libstdc++, logrotate, man, mod_perl, mod_ssl, ntp, openssh-server

Figure 3-18

- Unresolved Dependencies (Figure 3-18)
    - Click “Install packages to satisfy dependencies”
    - Click “Next”

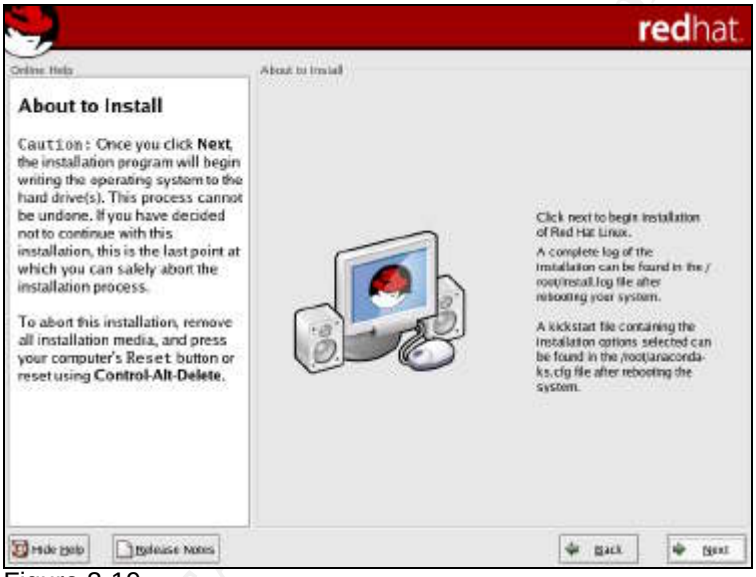


Figure 3-19

- About to Install (Figure 3-19)
    - Click “Next”

Figure 3-20

- Boot Diskette Creation (Figure 3-20)
  - Click “No, I do not want to create a boot diskette”
  - Click “Next”

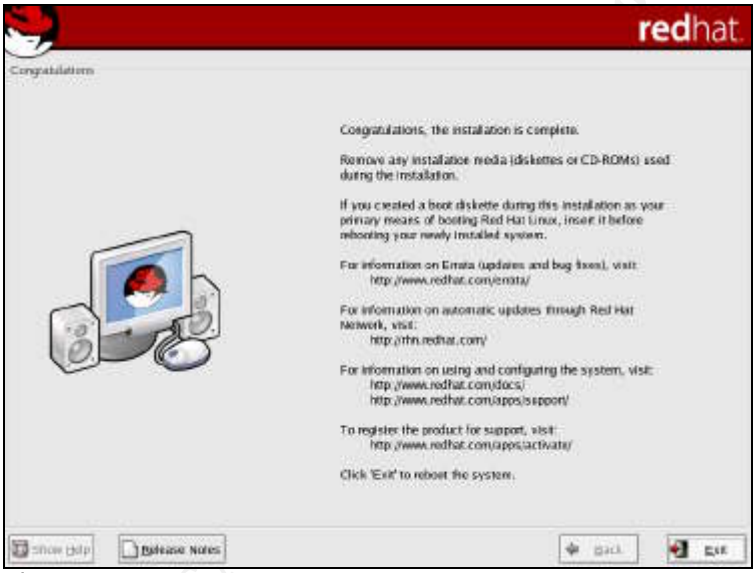


Figure 3-21

- Congratulations (Figure 3-21)
  - Click “Exit” to reboot

Appendix III

Test Lab Hardware

Engine
Dell Precision 420 Workstation

- Dual PIII 733MHz CPUs
- 512MB RAM
- 20GB HDD
- Onboard 3Com 3c9x 10/100 network interface card
- Generic video card
- Generic PS/2 keyboard
- Generic PS/2 mouse (optional)

Sensor (public and private)
Dell Precision 210 Workstation

- Single PIII 500MHz CPU
- 512MB RAM
- 17GB HDD
- Onboard 3Com 3c9x 10/100 network interface card
- Intel 10/100 EePro network interface card
- Generic video card
- Generic PS/2 keyboard
- Generic PS/2 mouse (optional)

Switches (all)

- Cisco Catalyst 3524XL

Appendix IV

Create an Archive Database

Setting up an archive database is very simple. The database structure is identical to the snort database, only with a different name. With that in mind, we can simply Follow the series of steps below to create the database on your engine server.

- Connect to mysql

```
[admin@ids-eng-01 admin]$ mysql -u root –p
Enter password: <password from step 2.3>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 3.23.54

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

- Create the snort_archive database

```
mysql> create database snort_archive;
```

- Connect to the snort_archive database

```
mysql> connect database snort_archive;
```

- Import the schema from /home/admin/packages/snort-2.0.1/contrib.

```
mysql> source /home/admin/packages/snort-2.0.1/contrib/create_mysql
        Query OK, 0 rows affected (0.01 sec)
        Query OK, 1 row affected (0.01 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.01 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.02 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 0 rows affected (0.01 sec)
        Query OK, 0 rows affected (0.00 sec)
```

```
        Query OK, 1 row affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 0 rows affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
        Query OK, 1 row affected (0.00 sec)
mysql>
```

- Assign privileges within the snort_archive database to the snort user

```
mysql> grant CREATE, INSERT, SELECT, DELETE, UPDATE on
snort_archive.* to snort;
Query OK, 0 rows affected (0.01 sec)

mysql> grant CREATE, INSERT, SELECT, DELETE, UPDATE on
snort_archive.* to snort@localhost;
Query OK, 0 rows affected (0.00 sec)
mysql>
```

- Exit mysql

```
mysql> exit
Bye
```

- Copy the /var/www/html/acid directory to /var/www/html/acid_archive

```
[admin@ids-eng-01 admin]$ sudo cp -R  /var/www/html/acid
/var/www/html/acid_archive
```

- Edit the new ACID configuration file

```
[admin@ids-eng-01 admin]$ sudo vi /var/www/html/acid_archive/acid_conf.php
```

- Change: $alert_dbname = "snort ";
  To: $alert_dbname = "snort_archive";

- Save the file

- Edit the apache configuration file

```
[admin@ids-eng-01 admin]$ sudo vi /etc/httpd/conf/httpd.conf
```

- Add the lines from Table 1-11 after the section that starts with " **<Directory "/var/www/html/acid_archive">**"

Table 1-11

```
<Directory "/var/www/html/acid_archive">
```

```
        AuthType Basic
        AuthName "ACID Console Archives"
        AuthUserFile /usr/lib/apache/passwords/passwords
        Require user admin
        AllowOverride None
</Directory>
```

- Save the file

- Restart the apache web server

```
[admin@ids-eng-01 admin]$ sudo /etc/init.d/httpd restart
Password:
Stopping httpd:                                [  OK  ]
Starting httpd:                                [  OK  ]
[admin@ids-eng-01 admin]$
```

In a web browser on a laptop or desktop connected to the management network, load the following URL:

http://10.13.10.101/acid_archive

You should be prompted for a username and password. Login in as admin, using the password from step 3.3. Proceed to initialize the snort_archive database, using step 4.4 as a guide.

The snort_archive database is now ready to start receiving data.

Last Updated: December 17th, 2017

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Security East 2018	New Orleans, LAUS	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS Amsterdam January 2018	Amsterdam, NL	Jan 15, 2018 - Jan 20, 2018	Live Event
Northern VA Winter - Reston 2018	Reston, VAUS	Jan 15, 2018 - Jan 20, 2018	Live Event
SEC599: Defeat Advanced Adversaries	San Francisco, CAUS	Jan 15, 2018 - Jan 20, 2018	Live Event
SANS Dubai 2018	Dubai, AE	Jan 27, 2018 - Feb 01, 2018	Live Event
SANS Las Vegas 2018	Las Vegas, NVUS	Jan 28, 2018 - Feb 02, 2018	Live Event
Cyber Threat Intelligence Summit & Training 2018	Bethesda, MDUS	Jan 29, 2018 - Feb 05, 2018	Live Event
SANS Miami 2018	Miami, FLUS	Jan 29, 2018 - Feb 03, 2018	Live Event
SANS Scottsdale 2018	Scottsdale, AZUS	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS London February 2018	London, GB	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Southern California- Anaheim 2018	Anaheim, CAUS	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Secure India 2018	Bangalore, IN	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Dallas 2018	Dallas, TXUS	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Brussels February 2018	Brussels, BE	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Secure Japan 2018	Tokyo, JP	Feb 19, 2018 - Mar 03, 2018	Live Event
Cloud Security Summit & Training 2018	San Diego, CAUS	Feb 19, 2018 - Feb 26, 2018	Live Event
SANS New York City Winter 2018	New York, NYUS	Feb 26, 2018 - Mar 03, 2018	Live Event
CyberThreat Summit 2018	London, GB	Feb 27, 2018 - Feb 28, 2018	Live Event
SANS London March 2018	London, GB	Mar 05, 2018 - Mar 10, 2018	Live Event
SANS San Francisco Spring 2018	San Francisco, CAUS	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Secure Osaka 2018	Osaka, JP	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Secure Singapore 2018	Singapore, SG	Mar 12, 2018 - Mar 24, 2018	Live Event
SANS Paris March 2018	Paris, FR	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS SEC460: Enterprise Threat Beta	OnlineCAUS	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced