



CompTIA Security+® Lab Series

Lab 2: Secure Network Administration Principles - Log Analysis

CompTIA Security+® Domain 1 - Network Security

Objective 1.2: Apply and implement secure network administration principles

Document Version: **2013-08-02**

Organization: Moraine Valley Community College

Author: Jesse Varsalone

Copyright © Center for Systems Security and Information Assurance (CSSIA), National Information Security, Geospatial Technologies Consortium (NISGTC)

The original works of this document were funded by the National Science Foundation's (NSF) Advanced Technological Education (ATE) program Department of Undergraduate Education (DUE) Award No. 0702872 and 1002746; Center for Systems Security and Information Assurance (CSSIA) at Moraine Valley Community College (MVCC).



This work has been adapted by The Department of Labor (DOL) Trade Adjustment Assistance Community College and Career Training (TAACCT) Grant No. TC-22525-11-60-A-48. The National Information Security, Geospatial Technologies Consortium (NISGTC) is authorized to create derivatives of identified elements modified from the original works. These elements are licensed under the Creative Commons Attributions 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

The Network Development Group (NDG) is given a perpetual worldwide waiver to distribute per US Law this lab and future derivatives of these works.

Contents

Introduction	3
Objective: Apply and implement secure network administration principles	3
Pod Topology	4
Lab Settings	5
1 Log Analysis in Linux Using grep	8
1.1 Using grep	8
1.2 Conclusion	15
1.3 Discussion Questions	15
2 Log Analysis in Linux Using gawk	16
2.1 Using gawk	16
2.2 Conclusion	22
2.3 Discussion Questions	22
3 Log Analysis in Windows Using find	23
3.1 Using find in Windows	23
3.2 Conclusion	29
3.3 Discussion Questions	29
References	30

Introduction

This lab is part of a series of lab exercises designed through a grant initiative by the Center for Systems Security and Information Assurance (CSSIA) and the Network Development Group (NDG), funded by the National Science Foundation's (NSF) Advanced Technological Education (ATE) program Department of Undergraduate Education (DUE) Award No. 0702872 and 1002746. This work has been adapted by The Department of Labor (DOL) Trade Adjustment Assistance Community College and Career Training (TAACCCT) Grant No. TC-22525-11-60-A-48. This series of lab exercises is intended to support courseware for CompTIA Security+® certification.

By the end of this lab, students will be able to parse log files within Linux and Windows for information pertinent to security events on their system. Students will perform administration on Linux and Windows machines and view the logs from these tasks.

This lab includes the following tasks:

- 1 - Log Analysis in Linux Using grep
- 2 - Log Analysis in Linux Using gawk
- 3 - Log Analysis in Windows Using find

Objective: Apply and implement secure network administration principles

You may have read articles online describing situations where someone's passwords were stolen and then used to gain access to an account in order to steal money. The use of strong passwords is critical to protecting your accounts, as well as data and resources within an organization.

grep –Stands for Global Regular Expression Print. The GREP utility allows you to search through a large number of files and folders for specified text.

gawk – The Linux/UNIX gawk command will allow you to display output in an easy to display human readable format. Typing **gawk –help** in Linux will display gawk options.

find – This command can be used within Linux and Windows. The find command in Windows will allow you to search for a specific string within a large group of values.

secure – This log file tracks SSH, or Secure Shell, connections. It provides information such as IP addresses, and date and time stamps. It also tracks other events related to security, such as the creation of new user accounts and new group accounts.

access_log – This log file tracks HTTP, or Hyper Text Transfer Protocol, connections. It provides information such as IP addresses, user agents, and date and time stamps.

Pod Topology

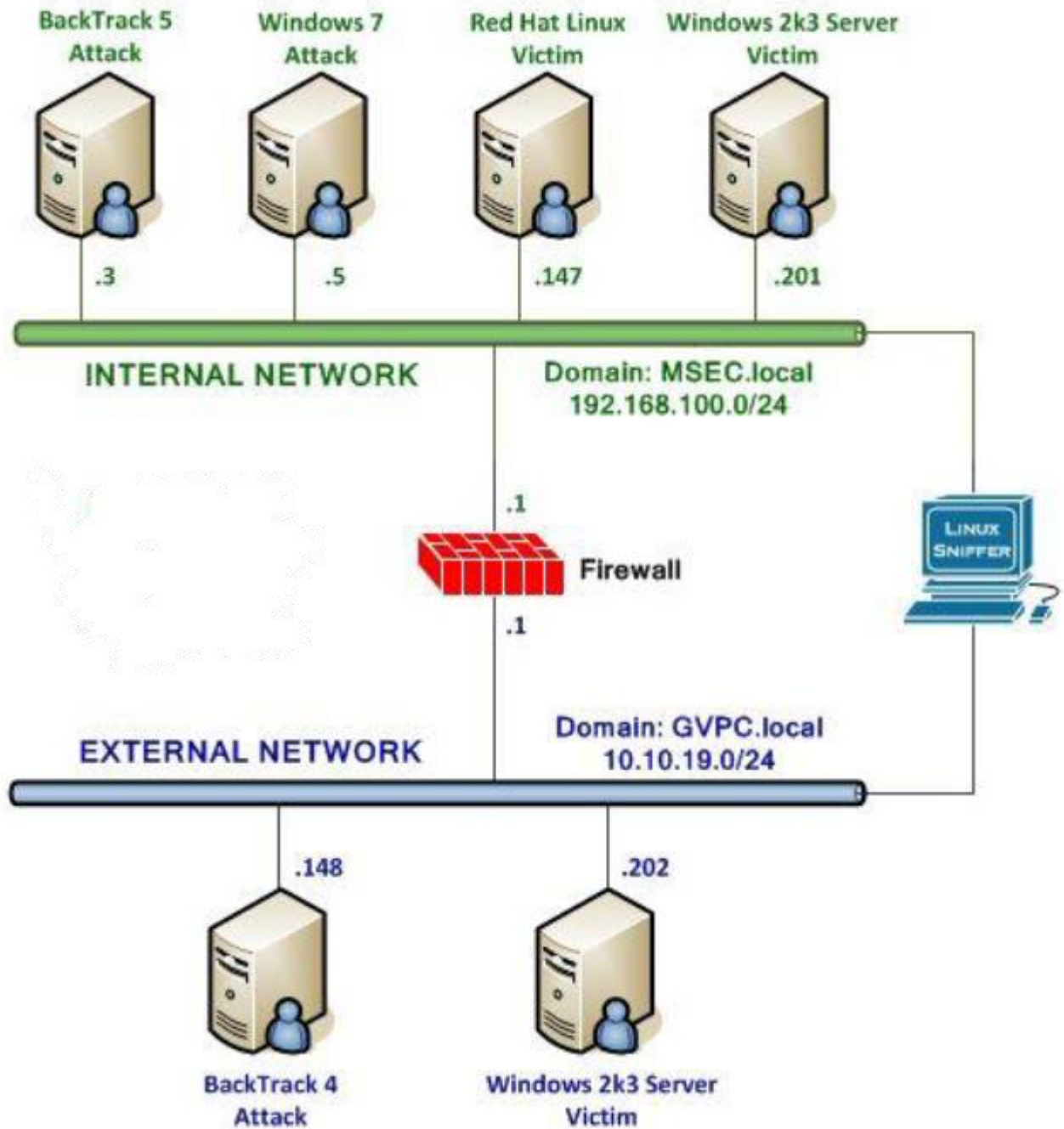


Figure 1: Topology

Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Required Virtual Machines and Applications

Log in to the following virtual machines before starting the tasks in this lab:

BackTrack 5 Internal Attack Machine	192.168.100.3
BackTrack 5 root password	password
Red Hat Enterprise Linux Internal Victim Machine	192.168.100.147
Red Hat Enterprise Linux root password	password
BackTrack 4 External Attack Machine	10.10.19.148
BackTrack 4 root password	password
Windows 2k3 Server External Victim Machine	10.10.19.202
Windows 2k3 Server administrator password	password

BackTrack 5 Internal Attack Login:

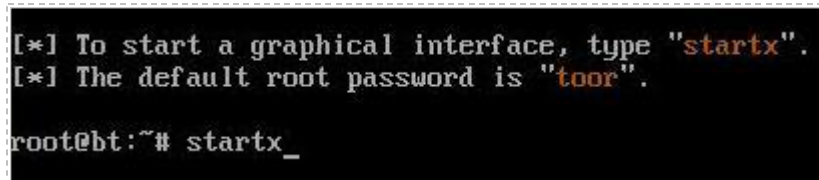
1. Click on the **BackTrack 5 Internal Attack** icon on the topology.
2. Type **root** at the **bt login:** username prompt and press **Enter**.
3. Type **password** at the **Password:** prompt and press **Enter**.



```
BackTrack 5 R1 - Code Name
bt login: root
Password: _
```

Figure 2: BackTrack 5 login

4. To start the GUI, type **startx** at the **root@bt:~#** prompt and press **Enter**.



```
[*] To start a graphical interface, type "startx".  
[*] The default root password is "toor".  
root@bt:~# startx_
```

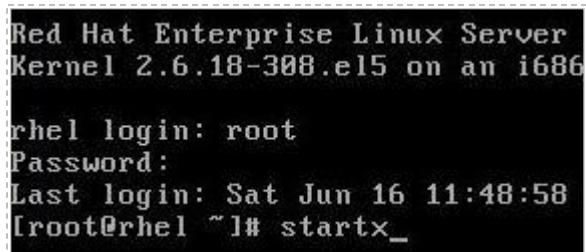
Figure 3: BackTrack 5 GUI start up

Red Hat Enterprise Linux Login:

1. Click on the **Red Hat Linux Internal Victim** icon on the topology.
2. Type **root** at the **rhel login:** prompt and press **Enter**.
3. Type **password** at the **Password:** prompt and press **Enter**.

For security purposes, the password will not be displayed.

4. To start the GUI, type **startx** at the **[root@rhe ~]#** prompt and press **Enter**.



```
Red Hat Enterprise Linux Server  
Kernel 2.6.18-308.el5 on an i686  
  
rhel login: root  
Password:  
Last login: Sat Jun 16 11:48:58  
[root@rhel ~]# startx_
```

Figure 4: RHEL login

BackTrack 4 External Attack Login:

1. Click on the **BackTrack 4 External Attack** icon on the topology.
2. Type **root** at the **bt login:** username prompt and press **Enter**.
3. Type **password** at the **Password:** prompt and press **Enter**.

For security purposes, the password will not be displayed.

4. To start the GUI, type **startx** at the **stroot@bt:~#** prompt and press **Enter**.

```
BackTrack 4 Beta bt tty1
bt login: root
Password:
Last login: Sat Jun 16 12:07:06 EDT
Linux bt 2.6.28.1 #2 SMP Wed Feb 4
++ WELCOME TO THE BACKTRACK LIVE CD

[*] To start Networking - "/etc/init
[*] To start KDE - "startx"
[*] To start FVWM - "bt4-crystal"

[*] http://www.remote-exploit.org/
stroot@bt:~# startx
```

Figure 5: BackTrack 4 login

:Windows 2k3 Server External Victim:

1. Click on the **Windows 2k3 Server External Victim** icon on the topology
2. Use the PC menu in the NETLAB+ Remote PC Viewer to send a **Ctrl-Alt-Del** (version 2 viewer), or click the **Send Ctrl-Alt-Del** link in the bottom right corner of the viewer window (version 1 viewer).
3. Enter the User name, **Administrator** (verify the username with your instructor).
4. Type in the password, **password**, and click the **OK** button (verify the password with your instructor).



Figure 6: Windows 2k3 login

1 Log Analysis in Linux Using grep

Within Network Administration, it is very important to check the system logs every day to monitor who is logging on and what type of activity is happening on a system. Log files can become extremely large, so tools like grep can be valuable in allowing the Network Administrator to filter values. Many log analysis jobs can be run using grep that will provide the Network Administrator with information on the status of a system.

1.1 Using grep

Open a terminal, use Nmap to identify open ports on a system, copy a website using curl and analyze by filtering with grep.

If you have already logged in and started the GUI interface, as described in the Lab Settings section, you may start immediately at Step 1.

Keep in mind that Linux commands are case sensitive. The commands must be entered exactly as shown, or errors will occur.

When starting the BackTrack 5 Internal Attack system, you must first enter in the username **root** followed by the password, **password**. At the initial start up screen, type the following command to start the GUI interface:
root@bt.~#startx.

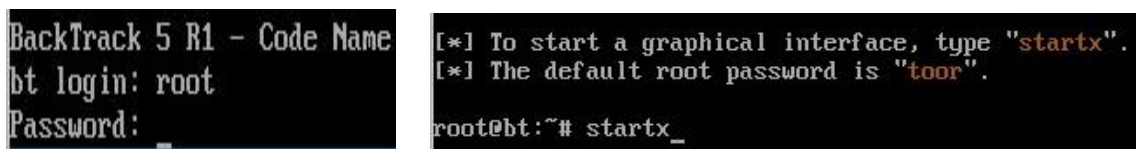


Figure 7: Linux Initial Startup Screens

1. Open a terminal on the BackTrack 5 Internal Attack system by clicking on the picture to the right of the word **System** in the task bar in the top of the screen.

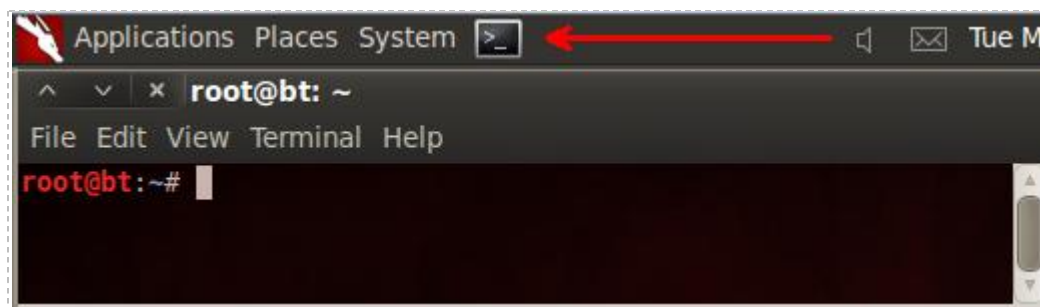


Figure 8: The Terminal Windows within BackTrack

Nmap, or Network Mapper, allows you to determine which TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) ports are open on a remote system. Zenmap is a GUI (Graphical User Interface) front-end for Nmap. Zenmap is packaged with Nmap.

2. Type the following command to launch the Zenmap application so you can perform a TCP Nmap scan of the Red Hat Linux Internal Victim:
`root@bt:~#zenmap`

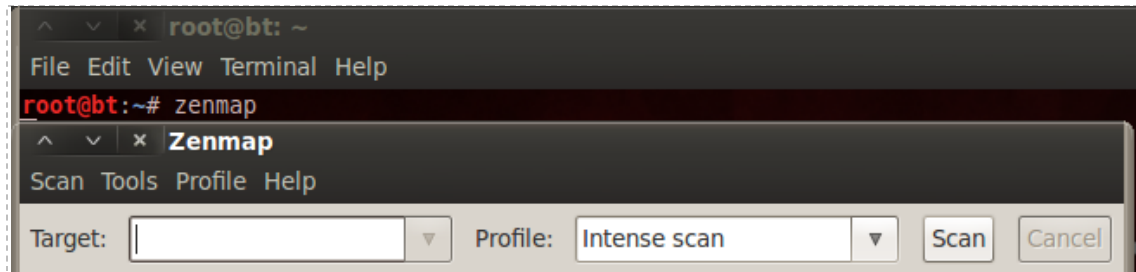


Figure 9: Zenmap can be Launched by Typing the Zenmap Command within the Terminal

3. In the target box, type the IP address of **192.168.100.147** (the Linux victim) and click the **Scan** button on the right.

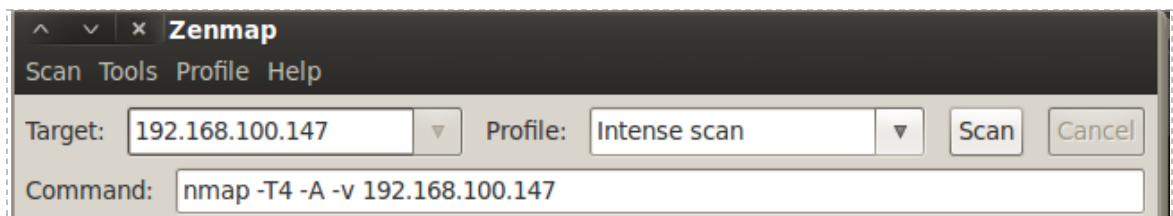


Figure 10: Entering the IP address of the Target Machine

Notice that the switches for Nmap are automatically added in the box directly below.

- After the scan is complete, click on the **Ports/Hosts** tab to display the open TCP ports.

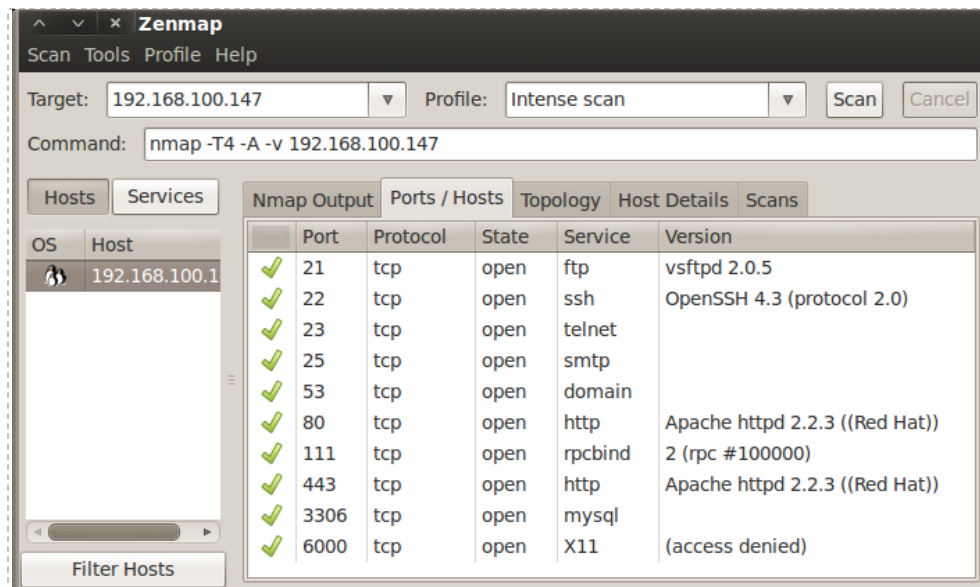


Figure 11: The Open TCP Ports on the Remote System

Notice that port 80 is open, which likely means the remote system is running a web server. The Zenmap scan indicates that the web server is Apache 2.2.3.

✓	80	tcp	open	http	Apache httpd 2.2.3 ((Red Hat))
---	----	-----	------	------	--------------------------------

Figure 12: The Remote System is running a Web Server

- Close the Zenmap tool by selecting **Scan** from the menu bar, and select **Quit**. Click **Close Anyway** if you receive a warning indicating that the scan is not saved.

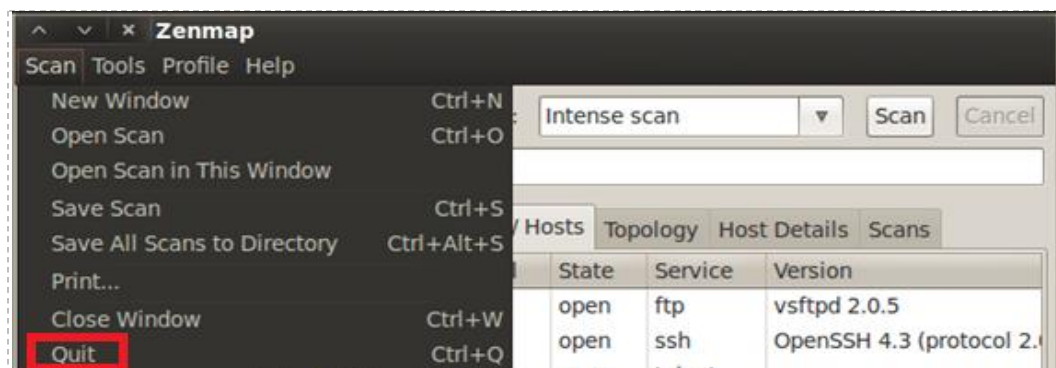


Figure 13: Closing Zenmap

Now that we know port 80 is open, we can attempt to connect to the target web site.

6. Open Firefox on the BackTrack 5 Internal Attack Machine, by performing the following steps:
Click **Applications** from the Menu bar, select **Internet**, then **Firefox Web Browser**.

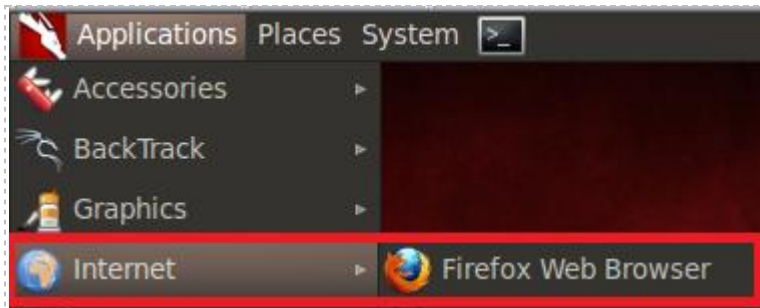


Figure 14: Opening Firefox on BackTrack

7. In the URL bar, type the address: <http://192.168.100.147> and press **Enter**.

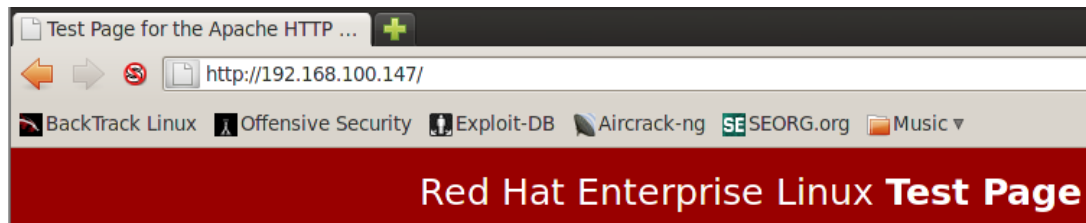


Figure 15: The Web Site of the Red Hat System

The test page likely indicates that the web site has not been configured. Close Firefox. Although you can view the HTML code of a web page in Firefox, there is also a Linux utility called **curl**, which stands for Client Uniform Resource Locator.

8. **Curl** can be used to make a copy of the website. On the BackTrack 5 Internal Attack terminal, type: `root@bt:~#curl http://192.168.100.147`

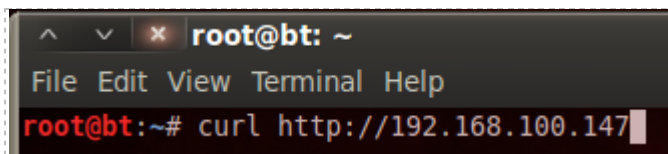


Figure 16: The curl command

The output from running the **curl** command will look similar to that below:



Figure 17: The Output from the curl command

Since the results from **curl** are large, you will find it helpful to filter them using the **grep** command. Notice the use of the pipe (|) symbol in the next step, it provides a useful way to link the output of several commands together. It is located directly above the **Enter** key on the keyboard and is generated using the **Shift+** key combination.

9. On the BackTrack 5 Internal Attack terminal, type the following to view HTML code and look for the word **test**:

```
root@bt:~#curl http://192.168.100.147 | grep test
```



Figure 18: Using grep to filter the results for the word test

The word **test** is highlighted in red within the paragraph of the HTML text that contains the word.

The Apache Server keeps records of the connections made to the website, including:

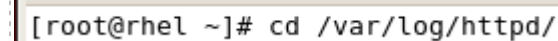
- IP addresses
- User Agents
- Date/Time Stamps

The **access_log** is located in the **/var/log/httpd** directory and will have evidence of:

- The scan of the target website with Zenmap
- The connection made with Firefox
- The connection made with the **curl** command

10. Switch over to the Red Hat Enterprise Linux Internal Victim Machine. To view the **access_log**, type the following command on the Red Hat system:

```
[root@rhel ~]# cd /var/log/httpd
```

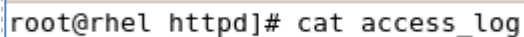


```
[root@rhel ~]# cd /var/log/httpd/
```

Figure 19: Switching to the Directory where the access_log is located

11. To view the connections in the log file, type the following command:

```
[root@rhel httpd]# cat access_log
```



```
root@rhel httpd]# cat access_log
```

Figure 20: Using the cat command to view the access_log

The results will appear similar to the results in the picture below.



```
[root@rhel httpd]# cat access_log
192.168.100.3 - - [28/May/2012:12:41:01 -0400] "GET / HTTP/1.0" 403 3985 "-" "-"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET / HTTP/1.1" 403 3985 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /robots.txt HTTP/1.1" 404 314 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /sdk/../../../../../../../../etc/vmware/hostd/vmInventory.xml HTTP/1.1" 404 336 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "OPTIONS / HTTP/1.1" 200 - "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /sdk/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/etc/vmware/hostd/vmInventory.xml HTTP/1.1" 404 336 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /favicon.ico HTTP/1.1" 404 315 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:44:11 -0400] "GET / HTTP/1.1" 403 3985 "-" "Mozilla/5.0 (X11; Linux i686; rv:5.0.1) Gecko/20100101 Firefox/5.0.1"
192.168.100.3 - - [28/May/2012:12:44:33 -0400] "GET / HTTP/1.1" 403 3985 "-" "curl/7.19.7 (i486-pc-linux-gnu) libcurl/7.19.7 OpenSSL/0.9.8k zlib/1.2.3.3 libidn/1.15"
192.168.100.3 - - [28/May/2012:12:44:40 -0400] "GET / HTTP/1.1" 403 3985 "-" "curl/7.19.7 (i486-pc-linux-gnu) libcurl/7.19.7 OpenSSL/0.9.8k zlib/1.2.3.3 libidn/1.15"
```

Figure 21: The access_log file

In Linux, the **access_log** file can be extremely long. The **grep**, or Global Regular Expression Print command can be used to filter the results of an access log or other output.

12. Type the following to filter the **access_log** file for the word **nmap** using **grep**:

[root@rhel httpd]# **cat access_log | grep Nmap**

```
[root@rhel httpd]# cat access_log | grep nmap
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET / HTTP/1.1" 403 3985 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /robots.txt HTTP/1.1" 404 314 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /sdk/../../../../../../../../etc/vmware/hostd/vmInventory.xml HTTP/1.1" 404 336 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "OPTIONS / HTTP/1.1" 200 - "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /sdk/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/etc/vmware/hostd/vmInventory.xml HTTP/1.1" 404 336 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
192.168.100.3 - - [28/May/2012:12:43:06 -0400] "GET /favicon.ico HTTP/1.1" 404 315 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
```

Figure 22: GREPing for the word nmap

13. Type the following to filter the **access_log** file for the word **Firefox** using **grep**:

[root@rhel httpd]# **cat access_log | grep Firefox**

```
[root@rhel httpd]# cat access_log | grep Firefox
192.168.100.3 - - [28/May/2012:12:44:11 -0400] "GET / HTTP/1.1" 403 3985 "-" "Mozilla/5.0 (X11; Linux i686; rv:5.0.1) Gecko/20100101 Firefox/5.0.1"
```

Figure 23: GREPing for the word Firefox

14. Type the following to filter the **access_log** file for the word **curl** using **grep**:

[root@rhel httpd]# **cat access_log | grep curl**

```
[root@rhel httpd]# cat access_log | grep curl
192.168.100.3 - - [28/May/2012:12:44:33 -0400] "GET / HTTP/1.1" 403 3985 "-" "curl/7.19.7 (i486-pc-linux-gnu) libcurl/7.19.7 OpenSSL/0.9.8k zlib/1.2.3.3 libidn/1.15"
192.168.100.3 - - [28/May/2012:12:44:40 -0400] "GET / HTTP/1.1" 403 3985 "-" "curl/7.19.7 (i486-pc-linux-gnu) libcurl/7.19.7 OpenSSL/0.9.8k zlib/1.2.3.3 libidn/1.15"
```

Figure 24: GREPing for the word curl

1.2 Conclusion

The access_log file within Linux provides information about connections to the server, including IP addresses, user agents, and date and time stamps. Log files can be extremely long and may contain a large amount of information about the connections made to the server. Linux utilities like grep can be used to filter the results of the file output.

1.3 Discussion Questions

1. Where is the access_log file located on a Linux system?
2. What is contained within the access_log file?
3. What does curl stand for?
4. How do you grep for the word nmap within the access_log?

2 Log Analysis in Linux Using gawk

While `grep` will allow you to filter the results of the file output, it will not really allow you to display the output differently. This is where `gawk` comes in; the Linux `gawk` command can be used to display the output of a text file in a more readable form.

2.1 Using gawk

Perform the following steps to generate security incidents on the Linux Victim system.

1. Open a terminal on the BackTrack 5 Internal Attack system by clicking on the picture to the right of the word **System** in the task bar in the top of the screen.

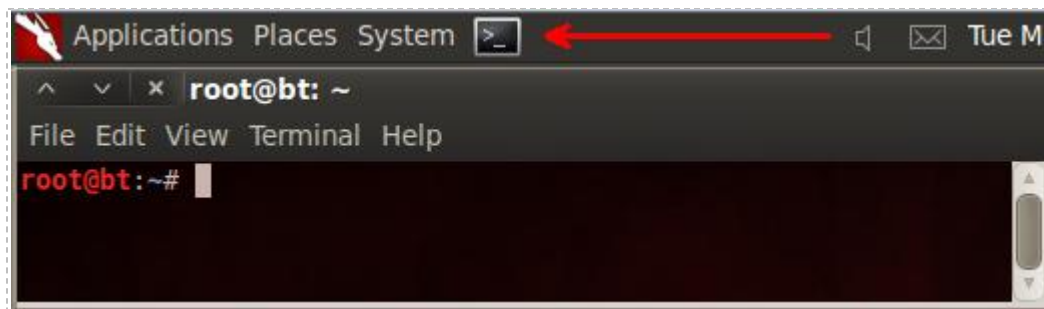


Figure 25: The Terminal Windows within BackTrack 5

2. Type the following command to SSH, or Secure Shell, to the remote system:
`[root@rhel ~]# ssh 192.168.100.147`
 - a. Type **yes** when asked "Are you sure you want to continue connecting (yes/no)?"
 - b. Type **password** for the password for root@192.168.100.147.

```
root@bt:~# ssh 192.168.100.147
The authenticity of host '192.168.100.147 (192.168.100.147)' can't be established.
RSA key fingerprint is 21:88:ba:44:07:d8:69:62:12:5f:49:f3:cc:ac:a3:24.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.100.147' (RSA) to the list of known hosts.
root@192.168.100.147's password:
Last login: Fri Jul 6 11:45:36 2012
[root@rhel ~]#
```

Figure 26: SSH to the Remote System

You should receive a message indicating your last login time on the system.

In order to create more security events, we will be creating the group **starwars**. We will create a total of three users. After creating each of the users and putting them in the group **starwars**, we will assign each user account a password. The chart below lists the users and passwords for our accounts in the starwars group.

Group: starwars	
User	Password
luke	son
vader	dad
yoda	green

3. Type the following command to add the group **starwars**:
[root@rhel ~]# **groupadd starwars**

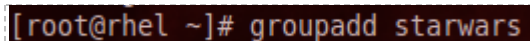


Figure 27: Adding the Group starwars

4. Type the following command to view the group file:
[root@rhel ~]# **cat /etc/group**

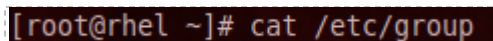


Figure 28: Viewing the Group File

If you scroll to the bottom of the group file, you will see the group that was created along with its corresponding unique group number.

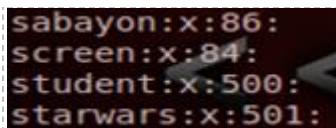


Figure 29: The group file

You can add users to the system in Linux by typing the **useradd** command. The **useradd** command will automatically create a directory with that user's name within the **/home** directory. When the user logs in, they will be placed into their directory within **/home**.

5. To add a user named **luke** and put him in the **starwars** group, type:
[root@rhel ~]# **useradd luke -g starwars**

```
[root@rhel ~]# useradd luke -g starwars
```

Figure 30: Adding the user luke

6. To add a user named **vader** and put him in the **starwars** group, type:
[root@rhel ~]# **useradd vader -g starwars**

```
[root@rhel ~]# useradd vader -g starwars
```

Figure 31: Adding the user vader

7. To add a user named **yoda** and put him in the **starwars** group, type:
[root@rhel ~]# **useradd yoda -g starwars**

```
[root@rhel ~]# useradd yoda -g starwars
```

Figure 32: Adding the user yoda

Next, we will give each user a password. We will use simple passwords for this exercise, but that should never be done on a production system. Avoid dictionary words because attackers can use programs like John the Ripper to crack short passwords or passwords that are found in a dictionary. Stick to passwords with a minimum of eight characters, uppercase and lowercase letters, and special characters. When you use a simple password with the **passwd** command, you will be warned that the password is a “BAD PASSWORD: it is WAY too short”. Retype the password again and it will be accepted.

For security reasons, the password will not be displayed when you type it.

8. Type the following to give **luke** a password:
[root@rhel ~]# **passwd luke**
Type **son** twice for the password.

```
[root@rhel ~]# passwd luke
Changing password for user luke.
New UNIX password:
BAD PASSWORD: it is WAY too short
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

Figure 33: Giving the user a Password

You should receive the message, *all authentication tokens updated successfully*.

9. Type the following to give vader a password:

```
[root@rhel ~]# passwd vader
```

Type **dad** twice for the password.

```
[root@rhel ~]# passwd vader
Changing password for user vader.
New UNIX password:
BAD PASSWORD: it is WAY too short
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

Figure 34: Giving the user a Password

You should receive the message, *all authentication tokens updated successfully*.

10. Type the following to give yoda a password:

```
[root@rhel ~]# passwd yoda
```

Type **green** twice for the password.

```
[root@rhel ~]# passwd yoda
Changing password for user yoda.
New UNIX password:
BAD PASSWORD: it is too short
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

Figure 35: Giving the user a Password

You should receive the message, *all authentication tokens updated successfully*.

When you perform administrative tasks that are directly related to the security on a Linux system, they will show up in the **secure** log in the `/var/log` directory. Examples of security incidents that will be recorded to the **secure** log include the following:

- Adding a user
- Logging on from a remote system
- Adding a group
- Changing a user's password.

11. To view the **secure** log, type the following command on the Red Hat system:

```
[root@rhel ~]# cd /var/log/
```

```
[root@rhel ~]# cd /var/log
```

Figure 36: Switching to the Directory where the secure log is located

12. To view the connections in the log file, type the following command:

```
[root@rhel log]# cat secure
```

```
[root@rhel log]# cat secure
```

Figure 37: Using the cat command to view the secure log

The results will appear similar to the results shown in the picture below.

```
May 28 16:21:06 rhel sshd[3690]: Received disconnect from 192.168.100.3: 11: disconnected by user
May 28 16:21:06 rhel sshd[3690]: pam_unix(sshd:session): session closed for user root
May 28 16:25:15 rhel sshd[3859]: Accepted password for root from 192.168.100.3 port 42236 ssh2
May 28 16:25:15 rhel sshd[3859]: pam_unix(sshd:session): session opened for user root by (uid=0)
May 28 17:42:31 rhel sshd[4247]: Connection closed by 192.168.100.128
May 28 17:45:02 rhel groupadd[4307]: new group: name=starwars, GID=501
May 28 17:59:48 rhel useradd[4648]: new user: name=luke, UID=501, GID=501, home=/home/luke, shell=/bin/bash
May 28 17:59:57 rhel useradd[4655]: new user: name=vader, UID=502, GID=501, home=/home/vader, shell=/bin/bash
May 28 18:00:05 rhel useradd[4660]: new user: name=yoda, UID=503, GID=501, home=/home/yoda, shell=/bin/bash
May 28 18:05:56 rhel passwd: pam_unix(passwd:chauthtok): password changed for luke
May 28 18:11:44 rhel passwd: pam_unix(passwd:chauthtok): password changed for vader
May 28 18:12:46 rhel passwd: pam_unix(passwd:chauthtok): password changed for yoda
```

Figure 38: The secure file on the Victim

Notice the file has information about new users and a new group created on the system, password changes, and contains information about incoming SSH connections.

13. Search for the instances of new user creation in secure by typing the following:

```
[root@rhel log]# cat secure | grep "new user"
```

```
[root@rhel log]# cat secure | grep "new user"
Jul  6 12:12:40 rhel useradd[12601]: new user: name=luke, UID=501, GID=501, home=/home/luke, shell=/bin/bash
Jul  6 12:12:58 rhel useradd[12611]: new user: name=vader, UID=502, GID=501, home=/home/vader, shell=/bin/bash
Jul  6 12:13:16 rhel useradd[12620]: new user: name=yoda, UID=503, GID=501, home=/home/yoda, shell=/bin/bash
```

Figure 39: GREPING for the new user events in the secure file.

With **grep**, the entire line containing the text you were searching for will be printed to the screen. When **gawk** is used, the default field separator is a space, so only the text to the left of the space you specify in the line will be displayed.. With the example below, only the text to the left of the first space will be printed to the screen.

gawk '{print \$1}' secure

Using the line below as an example, it would be the word **May** because that word is to the left of the first space.

May 28 17:59:48 rhel useradd[4648]: new user: name=luke, UID=501, GID=501, home=

To determine users created, use gawk to print values to the left of 6th, 7th, and 8th space.

Figure 40: The Space as a Field Separator

14. To determine the name of the new user created, we can use **grep** and **gawk**:

[root@rhel log]# gawk '{print \$6,\$7,\$8}' secure | grep "new user"

Figure 41: Using the GAWK command

2.2 Conclusion

The secure file in the /var/log directory will alert you to events related directly to the security on a Linux system, including account and password changes. The gawk command allows you to send specific output to the screen. The default delimiter for gawk is a space. As the secure log on a Linux system can become quite lengthy, the use of grep in conjunction with the gawk command will allow you to parse for certain events.

2.3 Discussion Questions

1. What are the results from typing the following command?
gawk -F= '{print \$2}' /var/log/secure
2. What are the results from typing the following command?
gawk '{print \$2}' /var/log/secure
3. What are the results from typing the following command?
(Type gawk --help to see the available switches and learn the purpose of "-F=")
gawk -F= '{print \$1}' /var/log/secure
4. What are the results from typing the following command?
gawk '{print \$1}' /var/log/secure

3 Log Analysis in Windows Using find

Windows also has many logs including the IIS, or Internet Information Services logs, which are text based logs. Neither gawk nor grep are part of Windows, although you can download third party versions. Windows has find, which will perform similar functions.

3.1 Using find in Windows

Open a terminal, perform a dictionary attack using xHydra, and use the find command to search the generated output.

1. Open a terminal on the BackTrack 4 External Attack system by clicking the picture to the left of Firefox in the task bar, located at the bottom of the screen.

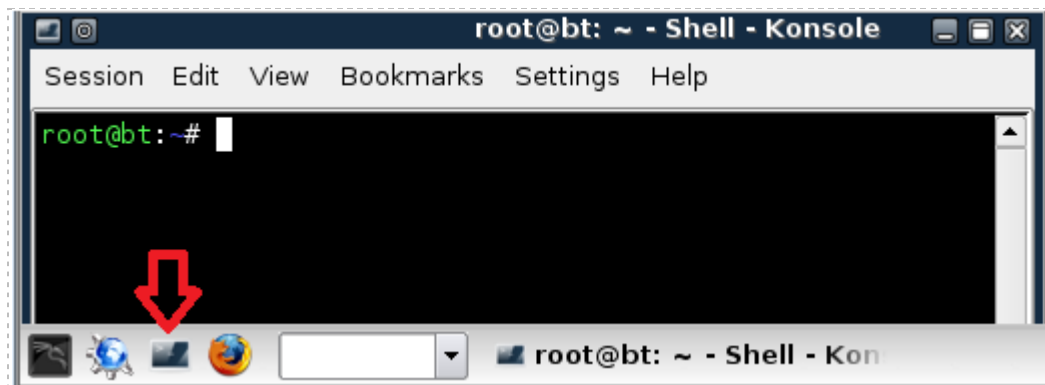


Figure 42: The Terminal Windows within BackTrack

The **xHydra** tool included with BackTrack will allow you to perform a dictionary attack against a remote system. We will be performing a dictionary attack on the FTP server with xHydra in order to generate a large amount of entries into the FTP log files.

2. Type the following command to launch the **xHydra** program on BackTrack 4 External Attack system:
`root@bt:~#xhydra`

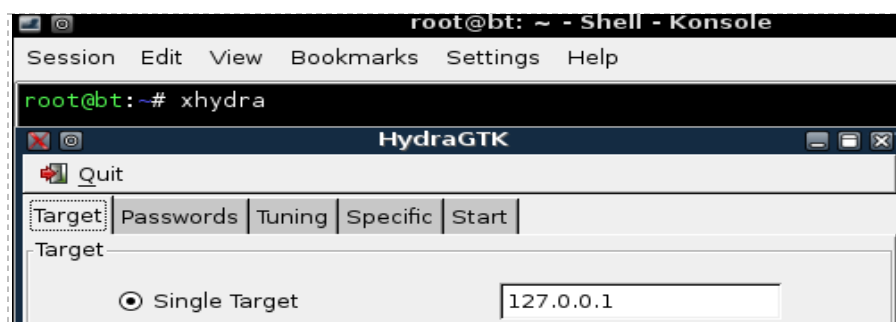


Figure 43: xHydra can be launched by typing the xhydra command within the Terminal

3. On the Target Tab, type **10.10.19.202**. Select **ftp** for the protocol.

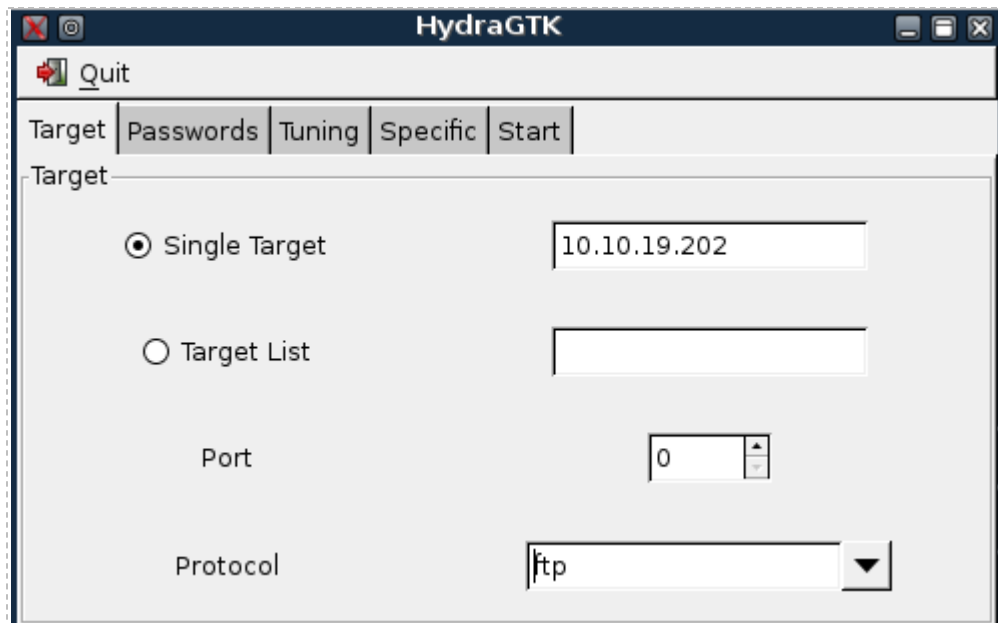


Figure 44: The Target Tab of xHydra

4. Click on the **Passwords** tab. Type **administrator** for the username. Under the password category, click on the **Password List** button.

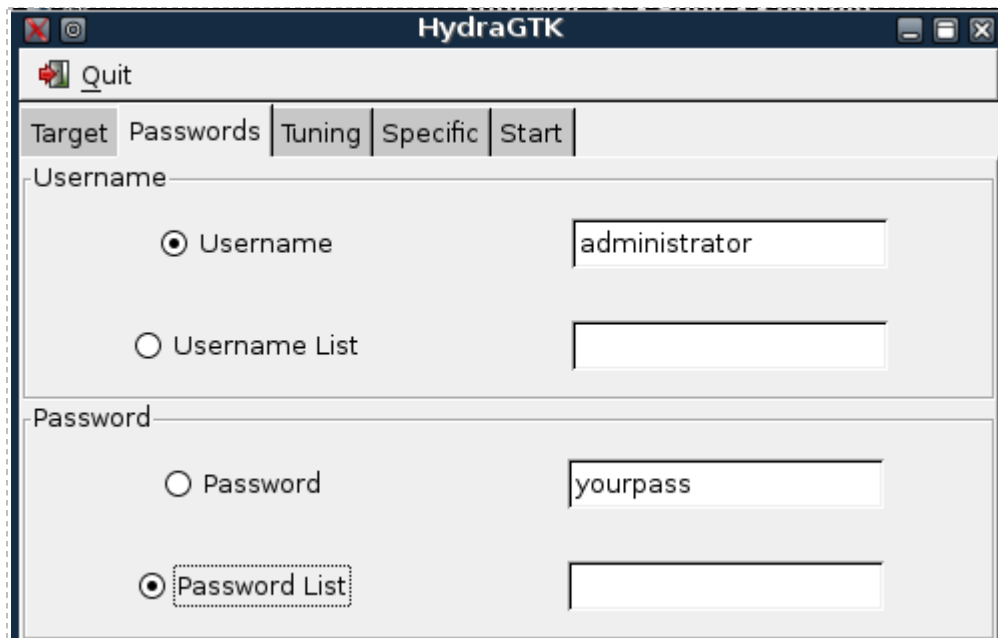


Figure 45: The Passwords Tab of xHydra

- Click in the white space to the right of the words **Password List** in xHydra.
Click the root directory, the click on **Wordlist.txt** and click the **Open** button.

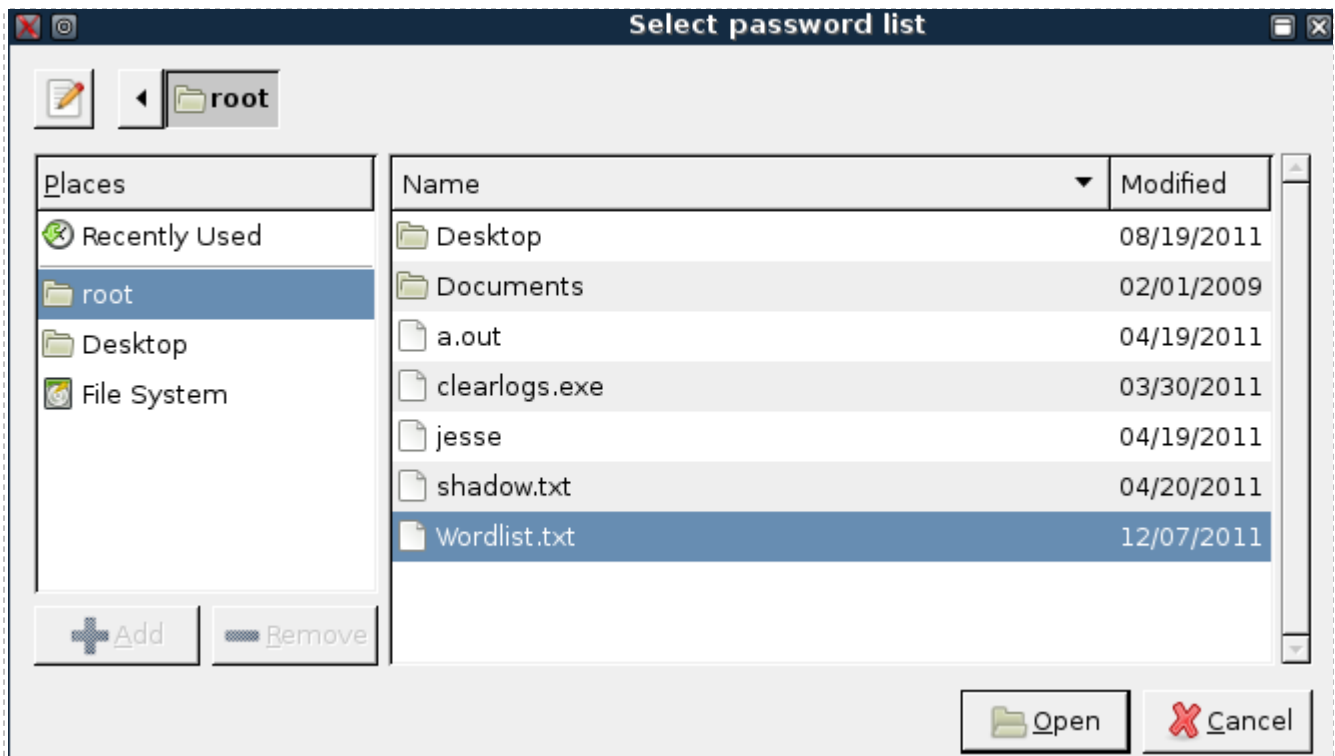


Figure 46: Selecting the Password File

/root/Wordlist.txt should now be listed in the Password List box.

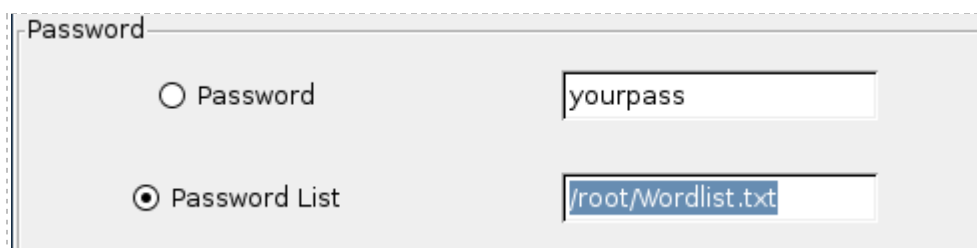


Figure 47: The Password List

- Click on the **Start** tab. At the bottom of the screen, verify that your xHydra program displays the options as shown in the picture below.

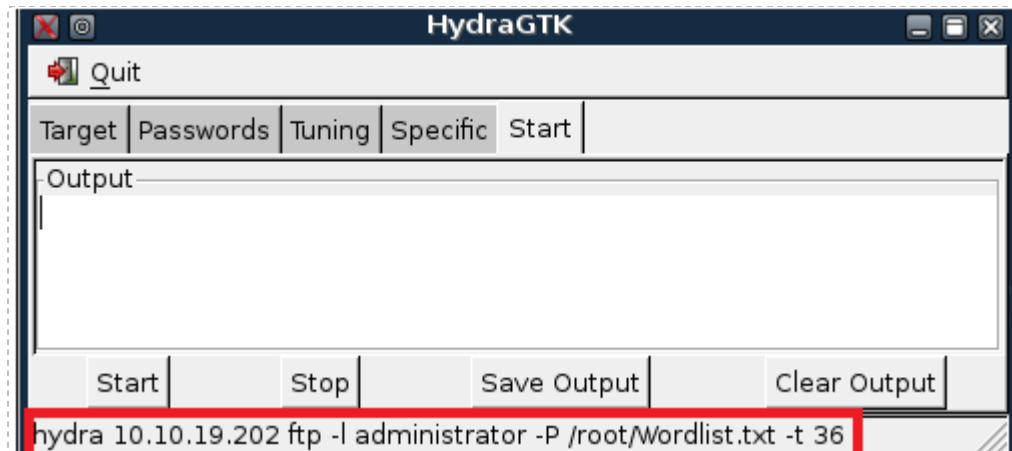


Figure 48: Verifying xHydra Options

- Click **Start**. It will take about 10-20 minutes to crack the administrator password.

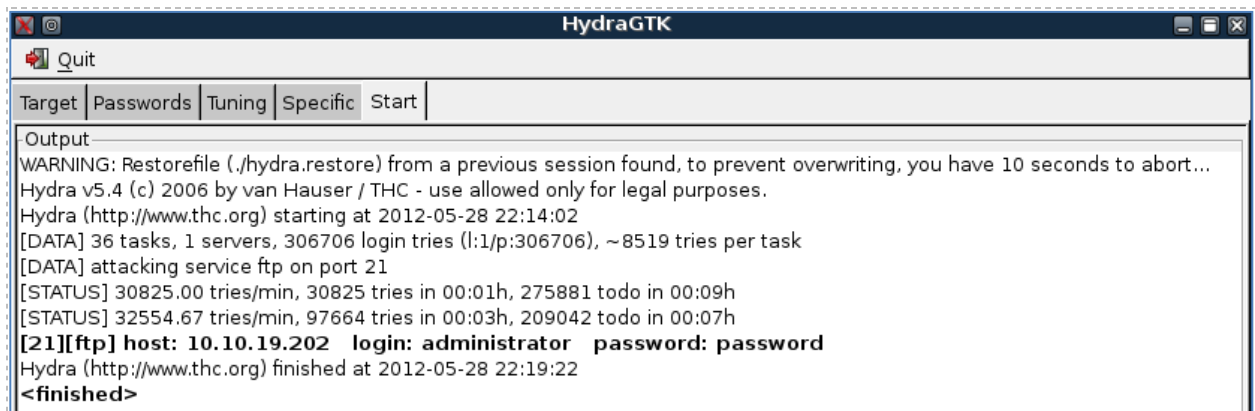


Figure 49: The password is cracked

8. Log on to the **Windows 2k3 Server External Victim Machine**. Use the PC menu in the NETLAB+ Remote PC Viewer to send a **Ctrl-Alt-Del** (version 2 viewer), or click the **Send Ctrl-Alt-Del** link in the bottom right corner of the viewer window (version 1 viewer). Log on with the username of **Administrator** and the password of **password**.

If you are already logged into the machine, you may skip this step.



Figure 50: Send Ctrl-Alt-Del to the Windows 2003 Server

9. On the Windows 2k3 Server External Victim Machine, open a command prompt by double-clicking on the **Command Prompt** shortcut located on the administrator's desktop.



Figure 51: Opening a Command Prompt

10. Navigate to the FTP Logfiles directory by typing the following command:
`C:\cd c:\Windows\System32\Logfiles\msftpsvc1`

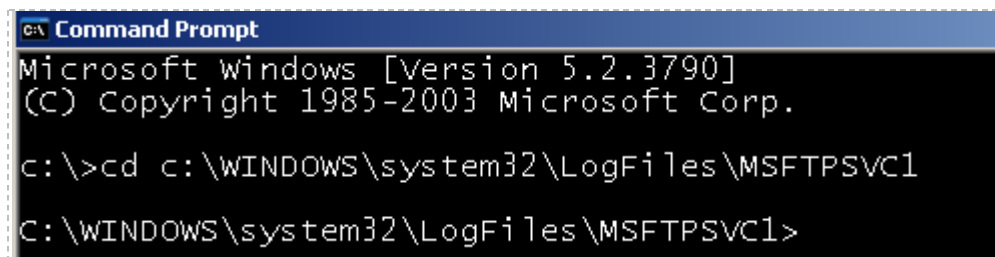


Figure 52: Navigating to the FTP Log files Directory

11. Type the following command to view all of the files in the directory:

C:\WINDOWS\system32\LogFiles\MSFTPSVC1>dir

```
C:\WINDOWS\system32\LogFiles\MSFTPSVC1>dir
Volume in drive C has no label.
Volume Serial Number is 7834-3125

Directory of C:\WINDOWS\system32\LogFiles\MSFTPSVC1

05/28/2012  10:14 PM    <DIR>          .
05/28/2012  10:14 PM    <DIR>          ..
01/18/2010  12:45 PM             440  ex100118.log
03/25/2012  12:59 PM            4,051  ex120304.log
03/25/2012  01:17 PM            2,794  ex120325.log
05/28/2012  10:19 PM       17,104,896  ex120529.log
               4 File(s)       17,112,181 bytes
               2 Dir(s)    1,433,866,240 bytes free

C:\WINDOWS\system32\LogFiles\MSFTPSVC1>
```

Figure 53: The FTP Log files

Notice how large today's log file is, because of the xHydra dictionary attack.

12. Type the following command to view the contents of the file:

C:\WINDOWS\system32\LogFiles\MSFTPSVC1>type exyymmdd.log

Use the most recent log file in the directory listing (see Step 11). The date, in the format of *yymmdd*, is part of the file name. This date will vary, depending on the day you perform this task. In this example, we would enter: **type ex120529.log | find "230"**

The results will appear similar to that of the results in the picture below.

```
02:19:21 10.10.19.148 [3]USER administrator 331 0
02:19:21 10.10.19.148 [35]USER administrator 331 0
02:19:21 10.10.19.148 [31]PASS - 530 1326
02:19:21 10.10.19.148 [13]PASS - 530 1326
02:19:21 10.10.19.148 [7]PASS - 530 1326
02:19:21 10.10.19.148 [22]PASS - 530 1326
02:19:21 10.10.19.148 [33]PASS - 530 1326
02:19:21 10.10.19.148 [10]PASS - 530 1326
02:19:21 10.10.19.148 [5]PASS - 530 1326
02:19:21 10.10.19.148 [32]PASS - 530 1326
02:19:21 10.10.19.148 [4]PASS - 530 1326
02:19:21 10.10.19.148 [18]PASS - 530 1326
02:19:21 10.10.19.148 [23]PASS - 530 1326
02:19:21 10.10.19.148 [17]PASS - 530 1326
02:19:21 10.10.19.148 [36]PASS - 530 1326
02:19:21 10.10.19.148 [12]PASS - 530 1326
02:19:21 10.10.19.148 [16]PASS - 530 1326
02:19:21 10.10.19.148 [1]PASS - 530 1326
02:19:21 10.10.19.148 [9]PASS - 530 1326
02:19:21 10.10.19.148 [8]PASS - 530 1326
02:19:21 10.10.19.148 [6]PASS - 530 1326
02:19:21 10.10.19.148 [14]PASS - 530 1326
02:19:21 10.10.19.148 [35]PASS - 530 1326
02:19:21 10.10.19.148 [11]PASS - 530 1326
02:19:21 10.10.19.148 [3]PASS - 530 1326
```

Figure 54: Today's FTP Log File

An incorrect password results in a 530 message. A 230 means the password was correct.

We can now use the find command to see if the user logged in successfully.

13. Type the following command to see if the attacker's login was successful:
C:\WINDOWS\system32\LogFiles\MSFTPSVC1> **type exyymmdd.log | find "230"**

Use the most recent log file in the directory listing (see step 11). The date, in the format of *yymmdd*, is part of the file name. This date will vary, depending on the day you perform this task. In this example, we would enter: **type ex120529.log | find "230"**

```
C:\WINDOWS\system32\LogFiles\MSFTPSVC1>type ex120529.log | find "230"  
02:19:21 10.10.19.148 [21]PASS - 230 0
```

Figure 55: Today's FTP Log File

We now know the date and time that the hacker successfully logged into the victim system.

14. Close all open windows and terminals.

3.2 Conclusion

The xHydra program allows an attacker to perform a dictionary attack against a variety of protocols, including FTP (File Transfer Protocol). A Windows system keeps logs of connection attempts in the C:\WINDOWS\system32\LogFiles\MSFTPSVC1 directory. These log files can be extremely long, so a user can use the find command to parse them.

3.3 Discussion Questions

1. What is xHydra?
2. Where are FTP Log files stored in Windows?
3. What is the code for a successful FTP login?
4. How can the find command be used to locate the number 230 within a logfile?

References

1. GREP man Pages:
<http://unixhelp.ed.ac.uk/CGI/man-cgi?grep>
2. Understanding /etc/shadow file The GNU Awk User's Guide:
<http://www.gnu.org/software/gawk/manual/gawk.html>
3. Windows Find Command:
<http://www.computerhope.com/findhlp.htm>
4. Files and Linux:
<http://www.irongeek.com/i.php?page=security/linuxlogs1>
5. THC-Hydra:
<http://www.thc.org/thc-hydra/>