GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

“WAREZ All My Disk Space?”

Responding to an Anonymous FTP Incident

Ryan C. Barnett

> **Wired Magazine – April 1997**
> Sunday morning, 7 a.m., somewhere in US Eastern Standard Time: Mad Hatter gets up, has a glass of Seagram's Ginger Ale and a cigarette, and checks his machine, which has been running automated scripts all night. He looks for errors and then reads his email. He has 30 messages from all over the world: some fan mail, a couple of flames, a few snippets of interesting information, three or four requests - some clear, some PGP-encoded. After a quick espresso and another cigarette, he surveys the contents of a few private FTP sites, filters through a bunch of new files, and then reroutes the good stuff to his newsreader. After breakfast with the family, another wave of automated scripts kicks in. The ISDN connection hums to life. A steady stream of bytes departs his machine 128 Kbps and vanishes into the ether. By the end of the day Mad Hatter, a ringleader of the software piracy group called the Inner Circle, will have poured 300 Mbytes of illegal "warez" onto the Internet. (1)

I. Executive Summary

The “WAREZ” underground is very real. They move huge amounts of pirated software all over the Internet, looking for storage space on any servers that will allow access. This process has been streamlined by proliferation of new technically sophisticated, yet extremely easy to use, scanning tools. In today’s World Wide Web, automated scanning of FTP sites is almost too easy.

> Writable areas can be very useful, but they have a dark side. Such writable directories *will* (notice that we didn't say *may*) be found and used by "the underground" on the Internet as storage space and distribution areas for illicit material; generally this means pirated software packages and pornographic image files.
>
> The folks who do this are amazingly well organized and hard to track down. They have their own communication mechanisms for telling each other about new sites - places they've found to store their stuff - without revealing who they are. (2)

Both the Security and Hacker/Cracker Communities agree that allowing Anonymous FTP access to a server greatly increases its vulnerability. It only becomes a matter of time until it is compromised or abused. It is, therefore, extremely important to realize the inherent security concerns associated with properly implementing this service.

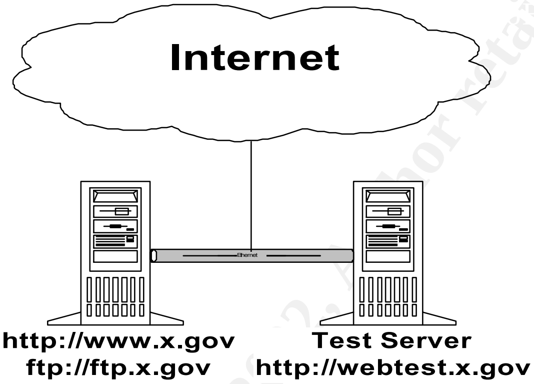
This paper will discuss a security incident that occurred on a U.S. Government Bureau’s web server. The incident itself was the result of an improperly configured ftp daemon, which allowed “Write” access to the ftp home directory. Identifying this misconfiguration during the incident response was rather basic. Tracking down “Why” this happened, on the other hand, turned out to be quite

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

complex. Unfortunately, the circumstances that lead to this security incident probably happen more often than are reported. It is the hope of this author that, by pointing out the mistakes made on the part of this System Administrator (SysAdmin), these same mistakes will not be repeated. At the very least, all Incident Handlers should address these situations.

There were two servers involved in this incident. One server was the live web server that hosted both Gov X's website and FTP archive and the other was the Backup/Test server.

A simplified network setup is for this incident is expressed below:



Each of the six phases of the Incident Handling (Preparation, Identification, Containment, Eradication, Recovery and Lessons Learned) process will be discussed, while focusing on this Anonymous FTP Incident. Most phases will conclude with a Synopsis section that serves as "Lessons Learned" from a post incident review of each respective category.

II. Preparation

The Preparation section will focus on the security components that were implemented prior to allowing Anonymous FTP access to the server.

- **Policy**

  Poor security policy was perhaps the biggest mistake made in the over all scope of this incident, excluding the misconfiguration of the ftp daemon itself. Repercussions of this deficiency were evident throughout the incident.

  The Bureau had an outdated Computer Security Incident Response Capability (CSIRC) handbook. This updated handbook had not been disseminated to the appropriate staff, which included the SysAdmin of the server and the immediate supervisors. This outdated document created obvious chaos when personnel identified a potential incident. What were the authorized steps to follow? Who should they call? Is the correct phone number listed? Didn't this contact person leave the Bureau six months ago? The list goes on and on. The main deficiency in the "Policy" area was not having an updated copy of the CSIRC handbook accessible to all authorized personnel.

SYNOPSIS – The updated CSIRC Handbook has since been supplied to all authorized personnel. The importance of having this resource available and containing accurate information was made abundantly clear during this Incident Response exercise.

- **Warning Banners**

The SysAdmin was relying on TCP-Wrappers (ftp://ftp.porcupine.org/pub/security/tcp_wrappers_7.6.tar.gz) to control both access to the server and to issue Warning Banners.

Before implementing Anonymous FTP, TCP-Wrappers was utilized as follows:

1. All ftp connections were run through TCP-Wrappers within the inetd.conf system. It was utilized to restrict ftp sessions by certain approved IP addresses. These IP's were for the developers who created and posted the web content. TCP-Wrappers issued the following Warning Banners:

```
************************************************** WARNING **************************************************************

THIS IS A BUREAU X COMPUTER NETWORK SYSTEM.  BUREAU X COMPUTER NETWORK SYSTEMS
ARE PROVIDED FOR THE PROCESSING OF OFFICIAL U.S. GOVERNMENT INFORMATION ONLY.
ALL DATA CONTAINED ON BUREAU X COMPUTER NETWORK SYSTEMS ARE OWNED BY BUREAU X,
AND MAY, FOR THE PURPOSE OF PROTECTING THE RIGHTS AND PROPERTY OF BUREAU X,
BE MONITORED, INTERCEPTED, RECORDED, READ, COPIED, OR CAPTURED IN ANY MANNER BY
AUTHORIZED SYSTEMS PERSONNEL.  THERE IS NO RIGHT OF PRIVACY ON THIS SYSTEM.
SYSTEMS PERSONNEL MAY GIVE TO LAW ENFORCEMENT OFFICIALS ANY POTENTIAL EVIDENCE
OF CRIME FOUND ON BUREAU X COMPUTER NETWORK SYSTEMS.  USE OF THIS SYSTEM BY ANY
USER, AUTHORIZED OR UNAUTHORIZED, CONSTITUTES CONSENT TO THIS MONITORING,
INTERCEPTION, RECORDING, READING, COPYING, OR CAPTURING AND DISCLOSURE.

************************************************** WARNING *************************************************************

            **** YOUR USE OF THIS SYSTEM IS BEING MONITORED ****
```

2. All other IP's were denied access and given a the following Refusal Banner:

```
*--Connection From Unknown Host Refused--*
*                                        *
* You Do Not Have Proper Access Privileges *
*  For This Host.  A System Administrator  *
*  Has Been Notified Of This Connection    *
*  Attempt And All Traffic Is Being Monitored  *
*                                        *
*****************************************************
```

© SANS Institute 2000 - 2002, Author retains full rights.

3. It is important to note that these Warning Banners were only visible to users if they accessed the server command line. If either a GUI FTP Client or a Web Browser was used, then the client did not visibly receive a Warning Message.

   - **Important -** Making a Warning Banner visible is of obvious importance, especially if there is any intent to potentially prosecute a violator.

Joining Anonymous FTP and TCP-Wrappers created some problems. The main roadblock to issuing the Warning Banners was the differing mechanisms by which these services control access. TCP-Wrappers applies access control based on "WHERE you come from" versus Anonymous FTP's access based on "WHO you are." The SysAdmin decided to configure the hosts.allow file of TCP-Wrappers to allow ftp attempts from anywhere on the Internet. The TCP-Wrappers "banner" system continued to issue the Warning Banner from above, however, this did not address the visibility of the banners from a Browser.

SYNOPSIS – The main question in this section is - How do you create banners that everyone would see? In order to issue a Warning Banner that all clients could see, an alternative FTP daemon should be used. This solution addresses access attempts from a GUI client, a browser or by commandline. An example of a more full featured alternative is the Washington University FTP daemon- (http://www.wuftpd.org). WU-FTPD allows the issuing of banners, either when a client logs into the archive or when the user changes into certain directories.

- **Proactive Techniques To Prevent An Incident**

  This section could easily be renamed "The Ironic Demise of Proactive Security." Although the SysAdmin implemented procedures to prevent improper access of the Anonymous FTP area, numerous events converged to negate every security feature implemented. All of the following procedures were implemented on the live webserver. They were not, however, completed and/or verified on the test server.

  1) The first step taken by the SysAdmin was to "Write" protect the FTP directories. He followed all of the standard steps to implement the correct permission rights to the directories. One of these steps was setting the permissions for the "/pub" directory:

```
# mkdir ~ftp/bin ~ftp/etc ~ftp/pub       Create needed directories.

Set up ~ftp/pub:
# chown root.wheel ~ftp/pub              Make sure root owns the directory.
# chmod 555 ~ftp/pub                     Make directory writable by nobody.
                                                                      (3)
```

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

SYNOPSIS- To ensure the proper implementation of Anonymous FTP, the SysAdmin and his Team Lead decided to first implement this service on the test server. Once the configuration was verified, it was then implemented to the live webserver.

To further ensure that the SysAdmin understood the security concepts, the Team Lead decided to issue a "Security Challenge". The Team Lead changed some of the security features of the implementation on the test server. The SysAdmin then had to identify and correct each misconfiguration. Some of the changes to the configuration included:

- Adding the user "anonymous" back to the ftpusers file
- Removing the "ls" binary from the ~ftp/bin directory
- Putting one encrypted password into the ~ftp/etc/passwd file
- Changing the permissions on the ~ftp/pub directory to 755.

The SysAdmin then went to work troubleshooting all of the configuration problems. It was assumed by the Team Lead that all of the changes were fixed. Can you guess which misconfiguration was not corrected or verified? Even though this exercise was completed on the test server, this error eventually came back to haunt the SysAdmin later in the incident.

2) Put a file quota on the ftp user, to limit the total number of bytes that can be received. (Alternatively, locate the anonymous FTP directory on an isolated partition.)

The next step in protecting the FTP archive was to set a quota limit on the ftp user. Normal implementation of Anonymous FTP includes the creation of a system user with the name "ftp". All users of Anonymous FTP are essentially coming into the system as this user. Without going into every aspect of adding user quotas, the final step is to edit the quota.user file. This file is located on the partition that the user quota will enforce. By issuing the "edquota ftp" command, you can edit this file through a normal text editor:

```
Quotas for user ftp:
/data: blocks in use: 20652, limits (soft = 27000, hard =
30000)
        inodes in use: 1070, limits (soft = 0, hard = 0)
```

The screenshot above shows what the quota restrictions were for the user "ftp" at the time of the abuse. Do you see any problems?

SYNOPSIS- The SysAdmin had the intention of implementing a sound security practice, much like the "Security Challenge" mentioned in the previous section. However, there was poor execution of the desired task.

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The implementation of user quotas with Anonymous FTP is very different from normal usage. Normal implementation grants users additional disk space for future use. In this case, we want to make sure that NO additional files are added to the archive. We, therefore, need to set the hard and soft limits to match exactly what the ftp user currently owns. After issuing the “edquota ftp” command, “you can add hard and soft limits for the user’s total disk space and inode space (total number of files). Setting a quota to 0 disables that quota.” (4)

The quota that was set for the ftp user above shows that both the hard and soft limits were not set correctly. Notice the inode setting in the previous example. Both the hard and soft limits were set to 0, which disabled that quota. If the desired result is to deny all new files to the FTP archive, the correct edquota should have been:

```
Quotas for user ftp:
/data: blocks in use: 20652, limits (soft = 20652, hard =
20652)
        inodes in use: 1070, limits (soft = 1070, hard =
1070)
```

This user quota setting would have restricted the FTP archive to disallow all additional files.

3) Use the “find” command to search for newly created files. This was the final security measure that the SysAdmin put in place. By using the “cron” scheduling system, the SysAdmin created an entry that periodically searched the ftp user’s home directory for newly created files. An initial search was run against the FTP archive to get a directory listing of the entire site. This was the base listing (base.file) of the archive. The ftp check “find” command piped it’s new output to a different file (ftptest.file). These two files were then compared with the “diff” command to show any new files.

```
www.x.gov>find /data/ftpdoc/ -name "*" > base.file
www.x.gov>find /data/ftpdoc/ -name "*" > ftptest.file
www.x.gov>diff base.file ftptest.file
1d0
< /data/ftpdoc
7a7
> /data/ftpdoc/intruder.file
```

SYNOPSIS- To summarize the configuration of the two servers in this incident:

- Live Server - Anonymous FTP is enabled. The /pub directory is not Writable, there is a user quota (although a bit high) and there is a cron job that looks for newly created files. This server is technically secure.

© SANS Institute 2000 - 2002, Author retains full rights.

- Test Server – Anonymous FTP is enabled. It has a “Writable” /pub directory (the SysAdmin missed this misconfiguration during the Security Challenge), a user quota (that is also set too high) and there is no cron job entry on this server. This server is not configured correctly for Anonymous FTP.

Now comes the most critical event in this incident.

On Tuesday, October 24, 2000, There was a hardware server problem that occurred on the live server. The SysAdmin of the boxes was offsite at the time so the Team Lead and another SysAdmin had to handle the situation. The live webserver needed to be taken offline to fix the problem, so it was, therefore, decided that the test server needed to go live as the web server.

I can hear SysAdmins screaming across cyberspace!

III. Identification

On the morning of Friday, October 27, 2000, the SysAdmin noticed something peculiar while reviewing the daily-automated system status e-mails. The following is a portion of the e-mail script output that runs the “df –k” command: (The /data directory holds the FTP archive)

```
df -k
Filesystem  1024-blocks       Used   Available Capacity  Mounted on
/dev/rz0h       1702466    1522585        9634   100%    /data

Filesystems over 75%
/dev/rz0h       1702466    1522585        9634   100%    /data
```

This was a big jump in % used on the /data partition from the previous day:

```
df -k
Filesystem  1024-blocks       Used   Available Capacity  Mounted on
/dev/rz0h       1702466    1257759      274460    83%    /data

Filesystems over 75%
/dev/rz0h       1702466    1257759      274460    83%    /data
```

This increase in capacity usage is what initially tipped off the SysAdmin that something might be wrong. He then looked through the remainder of the daily system emails and the syslog files for the webserver. The following entries were found in the daemon.log file: (Entries have been edited for content and length)

© SANS Institute 2000 - 2002, Author retains full rights.

```
Oct 26 12:28:03 www ftpd[747]: connection from dip.t-dialin.net at Thu Oct 26 12:28:03
2000
Oct 26 12:28:03 www ftpd[747]: <--- 220 www.x.gov FTP server (Digital UNIX Version
5.60) ready.
Oct 26 12:28:03 www ftpd[747]: command: USER anonymous^M
Oct 26 12:28:03 www ftpd[747]: <--- 331 Guest login ok, send ident as password.
Oct 26 12:28:03 www ftpd[747]: command: PASS XXXX
Oct 26 12:28:03 www ftpd[747]: <--- 230 Guest login ok, access restrictions apply.
Oct 26 12:28:03 www ftpd[747]: ANONYMOUS FTP LOGIN FROM dip.t-dialin.net,
id=anonymous@on.the.net
Oct 26 12:28:04 www ftpd[747]: command: CWD /net/pub/parent directory/^M
Oct 26 12:28:04 www ftpd[747]: <--- 250 CWD command successful.
Oct 26 12:28:04 www ftpd[747]: command: PWD^M
Oct 26 12:28:04 www ftpd[747]: <--- 257 "/net/pub/parent directory" is current directory.
Oct 26 12:28:09 www ftpd[747]: command: DELE hl1104.exe^M
Oct 26 12:28:09 www ftpd[747]: <--- 250 DELE command successful.
Oct 26 12:28:09 www ftpd[747]: command: DELE hlserver4104.exe^M
Oct 26 12:28:09 www ftpd[747]: <--- 250 DELE command successful.
Oct 26 12:28:29 www ftpd[747]: command: STOR flt-ra21.049^M
Oct 26 12:28:29 www ftpd[747]: <--- 150 Opening BINARY mode data connection for flt-
ra21.049 (216.203.143.141,1457).
Oct 26 12:33:00 www ftpd[747]: <--- 226 Transfer complete.
Oct 26 12:33:00 www ftpd[747]: store /net/pub/parent directory/flt-ra21.049 succeeded,
1502291 bytes.
Oct 26 12:33:00 www ftpd[747]: <--- 221 You could at least say goodbye.
Oct 26 12:33:00 www ftpd[747]: FTP LOGOUT, ftp
```

There are numerous concerns with these log files. First of all, are the entries concerning the successful deletion of two unknown files:

```
Oct 26 12:28:09 www ftpd[747]: command: DELE hl1104.exe^M
Oct 26 12:28:09 www ftpd[747]: <--- 250 DELE command successful.
Oct 26 12:28:09 www ftpd[747]: command: DELE hlserver4104.exe^M
Oct 26 12:28:09 www ftpd[747]: <--- 250 DELE command successful.
```

Not only shouldn't the command to "DELE" have succeeded, because the Delete command was supposed to have been disabled, but the fact that these files were present shows that there was prior "Write" access to this archive. The other issue is the successful uploading of an unknown file:

```
Oct 26 12:28:29 www ftpd[747]: command: STOR flt-ra21.049^M
Oct 26 12:28:29 www ftpd[747]: <--- 150 Opening BINARY mode data connection for flt-
ra21.049 (216.203.143.141,1457).
Oct 26 12:33:00 www ftpd[747]: <--- 226 Transfer complete.
Oct 26 12:33:00 www ftpd[747]: store /net/pub/parent directory/flt-ra21.049 succeeded,
1502291 bytes.
```

The successful uploading of this file was of obvious concern and further analysis was needed to determine the extent of this event.

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

To obtain additional evidence of the unauthorized file uploads, the SysAdmin telneted to the system and performed the steps outlined in the CERT Coordination Center's – Intruder Detection Checklist (http://www.cert.org/tech_tips/intruder_detection_checklist.html) (5). One of the steps suggests searching for "hidden files and/or directories":

> Look everywhere on the system for unusual or hidden files (files that start with a period and are normally not shown by 'ls'), as these can be used to hide tools and information (password cracking programs, password files from other systems, etc.). A common technique on UNIX systems is to put a hidden directory in a user's account with an unusual name, something like '...' or '.. ' (dot dot space) or '..^G' (dot dot control-G). Again, the find(1) program can be used to look for hidden files, for example:
>
> ```
>  find / -name ".. " -print -xdev
>
>  find / -name ".*" -print -xdev | cat –v
> ```
>
> Also, files with names such as '.xx' and '.mail' have been used (that is, files that might appear to be normal).

The SysAdmin went into the FTP home directory and searched for any hidden files.

```
www.x.gov>find /data/ftpdoc -name ".*" |more
/ftpdoc/net/pub/parent directory/.tagged
```

Two interesting pieces on evidence here:

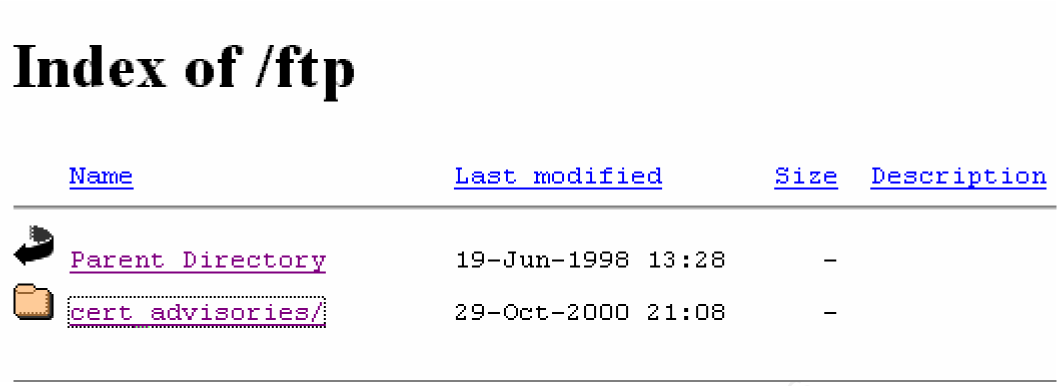
1) Obviously, the ".tagged" directory needed to be examined.
2) The parent folder of the .tagged directory called "parent directory" was also suspicious.

The directory naming was a pretty cunning move by the intruder. He created a new directory with a name resembling a system command – "CWD". If we look back at the previous daemon.log file, we can see the entries where the intruder went into this directory:

```
Oct 26 12:29:17 www ftpd[751]: command: CWD /net/pub/parent directory/^M
Oct 26 12:29:17 www ftpd[751]: <--- 250 CWD command successful.
```

From the SysAdmin's perspective, this could looks like normal user activity. It appears as if someone has moved into a new directory by clicking on the "Parent Directory" link. This movement, however, took the intruder into a directory below the current one instead of moving him up to the actual parent directory. Here is what a typical FTP archive directory looks like with the "Parent Directory" link :

© SANS Institute 2000 - 2002, Author retains full rights.



Index of /ftp

Name	Last modified	Size	Description
Parent Directory	19-Jun-1998 13:28	-	
cert advisories/	29-Oct-2000 21:08	-	

As this picture illustrates, the "Parent Directory" in not actually a directory, but a command to move into the parent directory. If the SysAdmin had been paying closer attention, he might have noticed this fact while reviewing the daemon.log file. The intruder moved into the "parent directory" and issued the CWD command:

```
Oct 26 12:29:17 www ftpd[751]: command: PWD^M
Oct 26 12:29:17 www ftpd[751]: <--- 257 "/net/pub/parent directory" is current directory.
```

If the intruder had actually clicked on the link for "Parent Directory", the results of the PWD command would have been "/net/pub" instead of "/net/pub/parent directory".

Examination of the ".tagged" directory showed the following (the –a flag was used with the ls command to show hidden files):

```
www.x.gov>pwd
/ftpdoc/net/pub/parent directory/.tagged
www.x.gov>ls -la
total 3
drwxr-xr-x   3 ftp system       512 Oct  26 12:21 .
drwxr-xr-x   3 ftp system       512 Oct  26 12:21 ..
drwxr-xr-x   2 ftp system       512 Oct  26 12:21 Lord_Raiden
```

Do you see a directory worth investigating? After going into the "Lord_Raiden" directory, the following files were found:

```
www.x.gov>ls -la
total 4130
drwxr-xr-x   2 ftp system       512 Oct  26 12:31 .
drwxr-xr-x   3 ftp system       512 Oct  26 12:21 ..
-rwxr-xr-x   1 ftp system   1502291 Oct 26 12:33 flt-ra21.049
-rwxr-xr-x   1 ftp system    933873 Oct 26 12:55 hl1104.exe
-rwxr-xr-x   1 ftp system   3276800 Oct 26 12:53 hlserver04.exe
```

At this point, the SysAdmin contacted the Team Lead and the CSIRC procedure was put into action. Unfortunately, as stated in the **Preparation** section, the

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

contact information in the CSIRC handbook was not accurate. The pager number for the Bureau's ISSO was listed incorrectly, as were several other phone numbers for key members of the CSIRC team. The appropriate personnel were eventually contacted, however, valuable response time was lost during this process.

Once the appropriate personnel were contacted, an Emergency Meeting was held.

IV. Containment

- **Assess The Situation**

  How did this happen? What is the extent of the compromise? These are some of the questions asked at the CSIRC meeting. The SysAdmin presented all of his findings to the team. It was then decided that this incident could not be classified as a classic "Hack" incident. A hack means that a system perameter was manipulated to achieve unintentional system access. It was clear that there was a misconfiguration of the FTP archive and that the "intruder" most likely did not receive a Warning Banner specifying the legalities of accessing this site. This incident, therefore, fell into a gray area between specified CSIRC incidents. These predefined incidents included: Malicious Logic Attacks, Probes and Reconnaissance Scans, etc… The closest classification that matched this incident was "Alteration/Compromise of Information", although technically, no data was altered but new data was created.

  The SysAdmin told the CSIRC team that he believed this was an FTP "WAREZ" incident. When asked why, the following evidence was submitted:

  - The daemon.log files for the previous three days were reviewed. The SysAdmin noticed that the intruder from dip.t-dialin.net had made some ftp connections on previous days. These connections were made late at night and there were numerous connection attempts made in a short period of time. Once the anonymous connection was successfully completed it was then dropped. These connection attempts appeared to be automated.

    There are countless open source tools and scripts that the WAREZ community uses to scan IP classes looking for Anonymous FTP sites – (http://david.weekly.org/code/ftpcheck.txt). These tools report all systems that have Anonymous FTP enabled and puts them into a list that the intruder can use at a later time. If the intruder had targeted Bureau X's server for attack, then other servers would not have been scanned. This scanning would increase the possibility of being detected. Upon reviewing the backup server's logs, it was indeed found that the intruder scanned the backup server for Anonymous FTP access.

© SANS Institute 2000 - 2002, Author retains full rights.

- The files that were uploaded to the FTP server had the Microsoft executable “exe” extension. These files, therefore, could not be executed on a Unix system. Logic suggests that there was no malicious intent to exploit the server directly utilizing these files. If an attack were planned for Bureau X’s webserver, reconnaissance would have shown that this was a Unix server. By putting an “exe” executable file on a Unix server, the intruder’s intention was assumed to be needing a place to store the file.

- These files are actually PC games. The SysAdmin used an Internet search website to find information about the new files. He went to the Lycos Fast FTP Search site – (http://ftpsearch.lycos.com/). Lycos returned the following results for the “hlserver4104.exe” file:

```
1 -rw-r--r-- 94.2M 2000 Oct 24 ftp.fasta.fh-dortmund.de
/gdi/halflife/server/win32/hlserver4104.exe
2 -rw-rw-r-- 94.2M 2000 Oct 25 ftp.gmd.de /people/cla/hl/hl-
newest/win/hlserver4104.exe
3 -r--r--r-- 94.2M 2000 Oct 26 ftp.phil.uni-erlangen.de
/pub/hl/patches/hlserver4104.exe
4 -r--r--r-- 94.2M 2000 Oct 25 ftp.task.gda.pl /vol/d33/ftp.extreme-
players.de/exp/hl/patches/hlserver4104.exe
5 -r--r--r-- 94.2M 2000 Oct 25 ftp.task.gda.pl
/vol/d33/ftp.gameaholic.com/pub/games/action/half-
life/official/hlserver4104.exe
```

  It appears that the “hlserver” is actually a server for the popular PC game “Half-Life” (http://www.half-life.com).

For all of the above reasons, this incident was classified as an FTP “WAREZ” incident.

- **Deploy A Team**

A CSIRC team was deployed to handle the situation. The team consisted of the SysAdmin, the Team Lead, two Bureau Forensic Analysts, Agents from two government agencies and myself.

- **Document What You Are Doing**

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Before anyone accessed the system, documentation of all CSIRC actions needed to be recorded. A mini-tape recorder was used to document every step taken by the CSIRC team. A tape recorder is the preferred method of documenting since it eases the task of having to document all events by hand. Things move extremely fast during an incident and it becomes easy to fall behind if one has to write everything by hand. Poor penmanship is also a great concern while documenting. Remember that all documentation is potentially evidence in a future legal case.

In addition to the tape recorder, the Unix Audit Logging subsystem was enabled. The auditing subsystem was originally turned off due to a lack of available logging space. Once the audit logging was enabled, all of the logs were sent across the network to the secure Linux box. By turning the audit logging on, there could be 100% verifiable evidence of all commands entered during the forensic analysis of the system. The functionality of the auditing subsystem on the server is described as:

> the administrator can use the audit subsystem to hold users accountable for their actions. The audit subsystem records every relevant security event that happens on the system (for example, each file open, file creation, login, and print job submitted).
>
> Each action is also stamped with an immutable audit ID (AUID) of the user who logged on, which allows all actions to be traced directly to a user. Users, by request to the system administrator, can use the audit trail to help recreate past events that affect the security of their accounts and data. (6)

- **Use Valid System Binaries**

  The CSIRC team "Jump Kit" contained many useful utilities including:
  - Hardware - Complete Red Hat Linux server tower, cables and hubs
  - Software - Forensic and Investigative software, CDROMs with compiled System binaries (ps, find, netstat, grep, etc…). These binaries were used instead of potentially "Trojaned" versions located on the Compromised system.

The CSIRC team was able to use Tripwire (www.tripwire.com) with two different collection databases. These databases were created previously by the SysAdmin. One database was of the entire system, while the other one was strictly for key system files. The first database was updated once a month and kept on a CDROM disk. Keeping this database (and all essential system binaries) on an un-modifiable medium is of great concern:

> The software and databases you use with them must be protected under all circumstances. If an intruder is able to penetrate your defenses and

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

> gain root access between scans, he or she can alter your programs and edit your comparison copies and databases to quietly accept whatever other changes are made to the system. For this reason, you want to keep the software and data on physically protected media, such as write-protected disks or removable disks. By interposing a physical protection between this data and any malicious hacker, you prevent it from being altered even in the event of a total compromise. (3)

The entire system database was used to validate the server.

- **Backup The System**

  The backup strategy for this incident was not a simple one. We did not want to shutdown the server and possibly corrupt system information that can be lost with a reboot. We also could not perform a data dump to tape because of a faulty tape drive on this server. It was, therefore, decided to copy all of the partitions to the secure Linux box via the network. We used a methodology similar to the one used by the Honeynet Project (http://project.honeynet.org/challenge/faq.htm).

  The link above explains the process that Honeynet makes images of a compromised system for their Forensic Challenge. Two of the tools used when we copied the partitions were the statically compiled "dd" Unix command and Netcat (http://www.l0pht.com/~weld/netcat/) both on the Forensic CDROM. The Honeynet Project has explained this disk imaging process:

> The following process was used to take the images, with minimum data pollution as a primary goal. We did not take the system down during the process. The following actions were taken while the system was still live.
> Mounted cdrom containing forensic analysis tools (all statically compiled).
> Used static binaries of dd(1M) and netcat(1M) from the cdrom to dd images of the hard drive to a trusted forensic system over the network. This is done by the following:
> Trusted System: Initiate a listening daemon on port 10000 of the trusted system using netcat:
>
> ```
> nc -l -p 10000 > honeypot.hda8.dd
> ```
>
> Compromised System: Copied each partition of the hard drive using dd(1), then piped the dd images of the drive over the network to the trusted system (192.168.1.1) listening on port 10000:
>
> ```
> /cdrom/dd bs=1024 < /dev/hda8 | /cdrom/nc 192.168.1.1 10000 -w 3
> ```
>
> This process was repeated for every partition on the hard drive, including swap. We now had a image of every partition on the trusted forensic system.

© SANS Institute 2000 - 2002, Author retains full rights.

Once these partitions were copied to the secure Linux system. We continued with the **Eradication** section.

V. Eradication

- **Determine The Cause**

Tracking down the cause for this particular incident was trivial. The SysAdmin had already determined, in his initial reconnaissance, that the FTP “/pub/” directory was “Writable” by the ftp user. The CSIRC team verified this by issuing the following command:

```
www.x.gov>pwd
/ftpdoc/net
www.x.gov>ls -l
total 1
drwxr-xr-x   3 ftp      system       512 Oct  26 14:20 pub
```

After verifying the permissions on this directory, a security assessment tool was used to verify the configuration of the entire system. If the ftp configuration was originally implemented incorrectly, what else might be misconfigured?” The tool selected for this task was the freeware utility “COPS” – by Dan Farmer (http://www.fish.com/cops/).

- **Perform Vulnerability Analysis**

The COPS utility was installed on the server and run against the entire system configuration. COPS validates a number of system configurations including:

- file, directory, and device permissions/modes.
- poor passwords.
- content, format, and security of password and group files.
- the programs and files run in /etc/rc* and cron(tab) files.
- existence of root-SUID files, their writeability, and whether or not they are shell scripts.
- a CRC check against important binaries or key files to report any changes therein.
- writability of users home directories and startup files (.profile, .cshrc, etc.)
- anonymous ftp setup.

COPS generated a report for the system that included the following information:

```
ATTENTION:
```

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
Security Report for Fri Oct 27 13:00:12 EST 2001
from host www.x.gov

ftp-Warning! /ftpdoc should be owned by root or root!
Warning!  /ftpdoc is _World_ writable! (*)
ftp-Warning! /ftpdoc should be mode 555!
```

After changing the permissions on the entire FTP archive, COPS was run again to verify the proper configuration of the FTP system.

- **Remove The Cause**

  Although removing the cause of the incident was elementary, deciding the exact new configuration of the webserver raised the following questions:

  - Should Anonymous FTP access to the server be denied entirely?

    The SysAdmin agreed with this proposal from the beginning. When first directed to allow Anonymous FTP by his superiors, he tried to convince them to spend the Time, Money and Resources to post the content in a different manner. Unfortunately, Anonymous FTP was the “easiest” way to make this content available. The “easy” solution prevailed.

  - If Anonymous FTP access was to be denied, how can the data be made available for the public?

    It was initially decided that creating the appropriate html archive for the data would be too time intensive. Allowing “directory listings” on the website was proposed as a potential solution. Directory listing will display all content in a directory if an “index.htm/html” page is not present. The output of a directory listing is similar to that of an Anonymous FTP listing within a browser. Although directory listing was an option, it is generally not a sound security practice to allow this type of access to your webserver. Too much information about your website structure could be given away to an intruder.

  - Should we set-up any traps to gather more information about the intruder?

    This idea was discussed quite extensively. More information was needed against the intruder to verify his location. The trick was getting more information about the intruder without him/her being notified and without leaving ourselves vulnerable. Additional tracking techniques are available in the **Appendix** section of this paper.

    All members of the CSIRC agreed with the following set-up:

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

- Allow the Anonymous FTP access to continue. Although the configuration was corrected to prevent new files from being uploaded.

- Delete the files previously uploaded (hlserver4104.exe and hl1104.exe) and replace them with fake files of the same name and size. Replacing the files removes any potentially malicious programs from the server while leaving a decoy in place. This technique camouflages the CSIRC's activities.

- TCP-Wrappers was then tweaked to immediately notify the CSIRC team if anyone from the dip.t-dialin.net domain tried to ftp to our server. The following Korn Shell script was used.

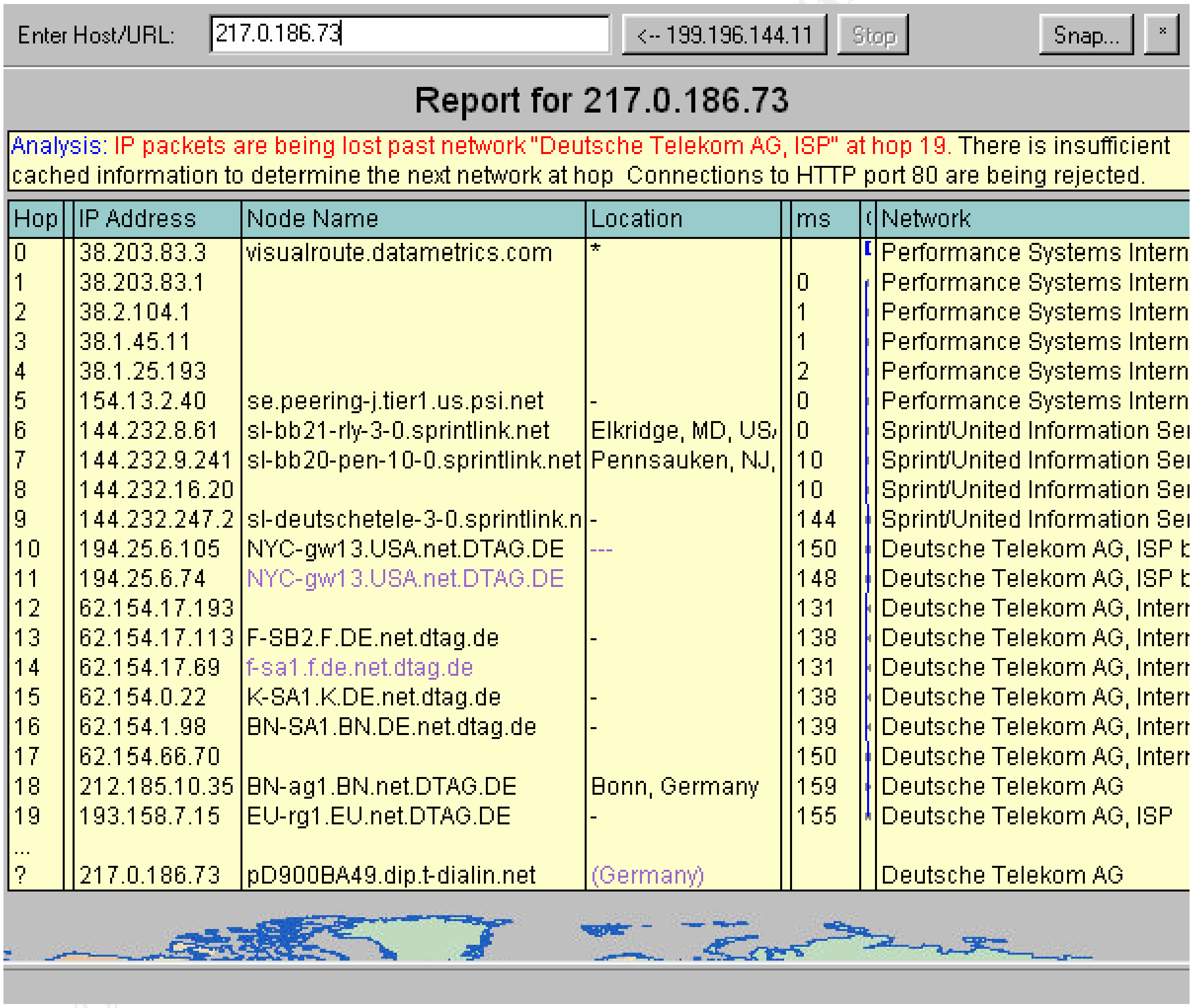
```
#!/bin/ksh
#
# Script launched by tcpd for intrusion detection purposes
#

USER=SysAdmin@www.x.gov
SRV=`echo $1 | cut -f1 -d.`
DATE=`date "+%a %b %e"`
TIME=`date "+%T"`
FINGER=`/usr/local/bin/safe_finger @$2`
MAIL=/usr/bin/mail

$MAIL $USER <<EOF
Subject: ### Intrusion Detection Alert From - `hostname` ###

You have received this alert because the host listed
above received a connection attempt from an unauthorized
client host.
The information below is the packet that was logged
and then dropped.

Date: $DATE
Time: $TIME
Source: $2
Destination: $3
Service: $SRV

--- Session Info ---

Finger Results - $FINGER
IP Of Client - $4
Client Info - $5
User Info - $6
EOF
```

© SANS Institute 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

- SWATCH (ftp://ftp.stanford.edu/general/security-tools/swatch/) was then used to monitor the daemon.log file and to notify us if anyone accessed the fake files.

- The IP address of the intruder in the daemon.log file was traced to find his actual geographic location. The CSIRC team went to the Visual Route website – http://www.visualroute.com and traced the IP to an ISP provider in Germany.



Report for 217.0.186.73

Analysis: IP packets are being lost past network "Deutsche Telekom AG, ISP" at hop 19. There is insufficient cached information to determine the next network at hop Connections to HTTP port 80 are being rejected.

Hop	IP Address	Node Name	Location	ms	(	Network
0	38.203.83.3	visualroute.datametrics.com	*			Performance Systems Intern
1	38.203.83.1			0		Performance Systems Intern
2	38.2.104.1			1		Performance Systems Intern
3	38.1.45.11			1		Performance Systems Intern
4	38.1.25.193			2		Performance Systems Intern
5	154.13.2.40	se.peering-j.tier1.us.psi.net	-	0		Performance Systems Intern
6	144.232.8.61	sl-bb21-rly-3-0.sprintlink.net	Elkridge, MD, US/	0		Sprint/United Information Se
7	144.232.9.241	sl-bb20-pen-10-0.sprintlink.net	Pennsauken, NJ,	10		Sprint/United Information Se
8	144.232.16.20			10		Sprint/United Information Se
9	144.232.247.2	sl-deutschetele-3-0.sprintlink.n	-	144		Sprint/United Information Se
10	194.25.6.105	NYC-gw13.USA.net.DTAG.DE	---	150		Deutsche Telekom AG, ISP b
11	194.25.6.74	NYC-gw13.USA.net.DTAG.DE		148		Deutsche Telekom AG, ISP b
12	62.154.17.193			131		Deutsche Telekom AG, Interr
13	62.154.17.113	F-SB2.F.DE.net.dtag.de	-	138		Deutsche Telekom AG, Interr
14	62.154.17.69	f-sa1.f.de.net.dtag.de		131		Deutsche Telekom AG, Interr
15	62.154.0.22	K-SA1.K.DE.net.dtag.de	-	138		Deutsche Telekom AG, Interr
16	62.154.1.98	BN-SA1.BN.DE.net.dtag.de	-	139		Deutsche Telekom AG, Interr
17	62.154.66.70			150		Deutsche Telekom AG, Interr
18	212.185.10.35	BN-ag1.BN.net.DTAG.DE	Bonn, Germany	159		Deutsche Telekom AG
19	193.158.7.15	EU-rg1.EU.net.DTAG.DE	-	155		Deutsche Telekom AG, ISP
...						
?	217.0.186.73	pD900BA49.dip.t-dialin.net	(Germany)			Deutsche Telekom AG

The "Whois" information for this ISP was obtained from Network Solutions –

Registrant: Deutsche Telekom Online Service GmbH (T-DIALIN2-DOM)
Waldstrasse 3
Weiterstadt, Germany D-64331 DE

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
Domain Name: T-DIALIN.NET

Record last updated on 12-May-2000.
Record expires on 10-Feb-2001.
Record created on 10-Feb-1999.
Database last updated on 19-Feb-2001 10:19:04 EST.

Domain servers in listed order:

DNS00.SDA.T-ONLINE.DE 195.145.119.62
DNS01.SDA.T-ONLINE.DE 195.145.119.189
DNS00.SUL.T-ONLINE.DE 194.25.2.123
DNS01.SUL.T-ONLINE.DE 194.25.2.124
```

- The real time alerts combined with the "Whois" information allowed the CSIRC team to coordinate further tracking efforts with the appropriate International authorities.

- **Locate A Clean Backup**

  Locating a clean backup was not difficult. Luckily, the SysAdmin had an appropriate backup strategy implemented. Since the system in question had already been verified with Tripwire, the archive backup was not needed. However, the tape archive was retained after putting the system back into production for emergency recovery purposes.

VI. Recovery

- **Restore The System**

  As previously stated, the system did not need to be restored from a tape backup. Restoring the system involved the previous steps outlined above.

- **Validate The System**

  The system was validated using a number of methods, including: running the system files against the Tripwire database, verifying the correct permissions on the FTP archive with COPS, and testing the permissions by visually confirming their settings.

- **Put Into Production**

  The server was put back into production with all of the alert mechanisms in place. It was then a matter of watching and waiting to see if the intruder would return.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

- **Monitor System**

On Saturday October 28, 2000, the intruder returned! The SysAdmin received the following email alert from TCP-Wrappers –

```
You have received this alert because the host listed
above received a connection attempt from an unauthorized
client host.
The information below is the packet that was logged
and then dropped.

Date: Sat. Oct 28
Time: 12:28:00
Source: www.x.gov
Destination: 0.0.0.0
Service: ftpd

--- Session Info ---

Finger Results -  [pD900BA49.dip.t-dialin.net]

IP Of Client – 217.0.186.73
Client Info - unknown
User Info - unknown

EOF
```

The intruder accessed the fake files to verify their existence and size and then disconnected.

```
Oct 28 12:28:09 www ftpd[3547]: command: PWD^M
Oct 28 12:28:09 www ftpd[3547]: <--- 257 "/net/pub/parent directory" is current directory.
Oct 28 12:28:09 www ftpd[3547]: command: SIZE hl1104.exe^M
Oct 28 12:28:09 www ftpd[3547]: <--- 213 191840
Oct 28 12:29:59 www ftpd[3547]: <--- 221 You could at least say goodbye.
Oct 28 12:29:59 www ftpd[3547]: FTP LOGOUT, ftp
```

Ten minutes later a SWATCH e-mail was received stating that one of the same files was accessed. This was peculiar since there was no TCP-Wrapper e-mail preceding this SWATCH e-mail. After reviewing the logs, it was discovered that a “different” intruder had accessed the same files!

This intruder was from a different ISP:

```
Oct 28 12:38:15 www ftpd[3577]: connection from hil-qbu-ppz-vty24.as.wcom.net at Sat
Oct 28 12:38:15 2000
Oct 28 12:38:15 www ftpd[3577]: <--- 220 www.x.gov FTP server (Digital UNIX Version
5.60) ready.
Oct 28 12:38:15 www ftpd[3577]: command: USER anonymous^M
Oct 28 12:38:15 www ftpd[3577]: <--- 331 Guest login ok, send ident as password.
Oct 28 12:38:16 www ftpd[3577]: command: PASS XXXX
Oct 28 12:38:16 www ftpd[3677]: <--- 230 Guest login ok, access restrictions apply.
```

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
Oct 28 12:38:16 www ftpd[3577]: ANONYMOUS FTP LOGIN FROM hil-qbu-ppz-
vty24.as.wcom.net, id=IEUser@
Oct 28 12:38:20 www ftpd[3577]: command: PWD^M
Oct 28 12:38:20 www ftpd[3577]: <--- 257 "/net/pub/parent directory" is current directory.
Oct 28 12:38:21 www ftpd[3577]: command: SIZE hl1104.exe^M
Oct 28 12:38:21 www ftpd[3577]: <--- 213 191840
Oct 28 12:38:22 www ftpd[3577]: RETR /net/pub/parent directory/hl1104.exe^M
Oct 28 12:38:23 www ftpd[3577]: <--- 150 Opening BINARY mode data connection for
hl1104.exe (206.175.101.24,1953) (191840)
Oct 28 12:39:39 www ftpd[3577]: <--- 221 You could at least say goodbye.
Oct 28 12:39:39 www ftpd[3577]: FTP LOGOUT, ftp
```

This information, along with the evidence gathered in the analysis of the log files, further confirmed the hypothesis that this was an FTP "WAREZ" incident. More than likely, this intruder obtained the location of these files from the original intruder and then attempted to retrieve them.

All of the appropriate evidence was handed over to the proper Bureau and Government Agency authorities. After a week or two of monitoring, Anonymous FTP was disallowed entirely on the server.

VII. Follow Up / Lessons Learned

- **Follow Up Report**

There are a many points that need to be made concerning how this incident happened. The following recommendations apply to all scenarios involving Internet Security not just this particular scenario –

- An updated and accurate CSIRC handbook needs to be issued to all appropriate employees. For example, not having an accurate handbook caused confusion and loss of valuable time when the SysAdmin first identified the intrusion.

- **Never** – use a backup server as a test box. Numerous reconfigurations create an indeterminate number of security holes. While a SysAdmin never intends to break this cardinal rule, the lack of monetary resources for an adequate number of servers often propagates this unsound practice.

- **Always** – double-check your security implementations, whether it is with automated security tools, such as COPS, or by having a colleague review the set-up. There is nothing more embarrassing than when an incorrectly implemented security measure is exploited.

- As a result of this incident, SWATCH has been implemented on all of the servers used for web services. This tool is very flexible and greatly reduces the amount of actual time spent reviewing logs files. This tool

© SANS Institute 2000 - 2002, Author retains full rights.

also allows for real-time notification versus reviewing the logs once a day in the morning. This allows for more prompt response times to future incidents.

The event described in following news article very well could have happened to this server if the abuse was not identified promptly:

> CNET News – Dec. 14 2000
> **Power play: Electric company hacked**
> The NIPC says an unnamed power company was turned into an Internet game server by unknown intruders. Unknown intruders have hijacked an electric company's servers, using its computers and the company's Internet connection to host and play games, the National Infrastructure Protection Center revealed Wednesday. The intruders used the hacked FTP site to store and play interactive games that consumed 95 percent of the organization's Internet bandwidth," NIPC said in a prepared statement. "The compromised bandwidth threatened the (company's) ability to conduct bulk power transactions. (8)

In closing, several circumstances served as a catalyst for the incident. However, the SysAdmin's diligence in reviewing the system status minimized the severity of the ftp misconfiguration.

VIII. Appendix – Tracking The Intruder

This process falls into a transitional area of responsibility for Incident handling. It is the responsibility of both the Incident Response team and the appropriate authorities to gather evidence. This issue was discussed among the CSIRC team quite extensively while responding to this Anonymous FTP incident. After conferring with the Agents from the Secret Service and the FBI, it was agreed that initial reconnaissance was needed to adequately categorize the incident at hand. Tracking the intruder can be a dicey practice. The goal in tracking an intruder is to gather as much information as possible about his real identity, without alerting him to your activities. This evidence could be critical to the successful prosecution of an intruder. As mentioned in the SANS "Incident Handling – Step By Step" Containment chapter:

> **Action 3.2.1 Avoid looking for the attacker with obvious methods.**
> A classic rookie error is to ping, nslookup, finger, telnet to, or in some other way, make contact with the suspected source of the attack (hours later).
> If your adversaries detect you trying to locate them, they may delete your file systems and break off the connection (for a while anyway) in an effort to cover their tracks.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The above section cautions against the practice of "Direct" reconnaissance against an intruder. Standard "pings" and "traceroutes" can alert an intruder to your queries and could be extremely dangerous. The best practice is to use anonymous Internet tools to perform these searches. The following websites can be utilized to conduct anonymous reconnaissance against an intruder:

- http://www.visualroute.com
- http://packetderm.cotse.com/cgi-bin/lookuptools
- http://users.rcn.com/rms2000/sleuth/index.htm
- http://www.anonymizer.com

These resources were used during the "Monitoring" phase of this incident. The Anonymizer website, which masks your real IP address when you access websites, was used to access one of the intruder's IP addresses obtained from the daemon.log file. As it turned out, a commercial company based in Texas owned this IP and was running a public website. The appropriate contact information for this website was obtained from Network Solutions (http://www.networksolutions.com) and the point of contact was notified of the problem. They were informed that files from their FTP archive were being transferred to Gov X's FTP archive. The CSIRC team received an interesting response from this person. "We have an FTP archive?" they replied. They seemed surprised! The company, at the time of writing this paper, no longer allows Anonymous FTP to their server.

The tools were quite effective in tracking the locations of the intruders. The key concept to utilizing these tools was that of timing. The CSIRC team was able to obtain fairly accurate information because of the alerting mechanisms that were implemented on the server. Once the team received an alert e-mail from TCP-Wrappers, they were able to immediately plug the host information into these web utilities and get real-time information. This was all accomplished while remaining conveniently anonymous.

IX. References

1. McCandless, David. "Warez Wars". April 1997
   URL - http://hotwired.lycos.com/collections/hacking_warez/5.04_warez_wars1.html (22 Jan. 2001)
2. Chapman, D. Brent & Zwicky, Elisabeth D. "Building Internet Firewalls". First Edition. November 1995
3. Spafford, Gene & Garfinkel, Simson. "Practical Unix & Internet Security". April 1996
4. Frisch, Aeleen. "Essential System Administration". 2nd Edition. 1995.
5. CERT Coordination Center. "Intruder Detection Checklist". July 20, 1999

© SANS Institute 2000 - 2002, Author retains full rights.

URL - http://www.cert.org/tech_tips/intruder_detection_checklist.html (02 Feb. 2001)

6. Digital Equipment Corporation "Digital Unix Security". March 1996 URL - http://tru64unix.compaq.com/faqs/publications/base_doc/DOCUMENTATION/V40D_HTML/AQ0R2DTE/TITLE.HTM
7. Spitzner, Lance. "The Honeynet Project" http://project.honeynet.org/
8. Lemos, Robert. "Power play: Electric company hacked". December 14 2000 URL - http://www.zdnet.com/zdnn/stories/news/0,4586,2665199,00.html#more_on (20 Jan. 2001)

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.