

## Re: IPSwitch, Inc. WS\_FTP Server

**Source:** <http://www.derkeiler.com/Mailing-Lists/securityfocus/bugtraq/2002-10/0380.html>

---

**From:** Alun Jones ([alun@taxis.com](mailto:alun@taxis.com))

**Date:** 10/25/02

Date: Fri, 25 Oct 2002 12:38:29 -0500

To: [bugtraq@securityfocus.com](mailto:bugtraq@securityfocus.com)

From: Alun Jones <[alun@taxis.com](mailto:alun@taxis.com)>

At 09:06 AM 10/25/2002, [dev-null@no-id.com](mailto:dev-null@no-id.com) wrote:

>{ *Overview* }

>

> *WS\_FTP v3.13 by IPSwitch, Inc., is vulnerable to the classic FTP*

> *bounce attack as well as PASV connection hijacking.*

This start makes me sceptical from the first. A report of old vulnerabilities known to exist in a protocol, and with existing workarounds and solutions. Why is this news?

Let me say right away that I'm not affiliated with IPSwitch, I've visited their offices once, and I'm the developer of a competitor to their WS\_FTP Server.

>{ *Impact* }

>

> *The FTP bounce vulnerability allows a remote attacker to cause the*  
> *FTP server to create a connection to any IP address on any TCP port*  
> *greater than 1024. Thus, the attacker can scan Internet addresses*  
> *anonymously along with any internal addresses that the FTP server has*  
> *access to. More information on this vulnerability can be found here:*  
> <http://www.cert.org/advisories/CA-1997-27.html>.

Um... yes. This is not news, and is common to all FTP servers, because those pesky users insist on an FTP server that – gasp – does what the RFC says constitutes an FTP server. That means it's got to handle a PORT command correctly (if it doesn't, it isn't an FTP server, by definition). The server should have a feature to require that all PORT commands be on the same IP address that connected to the server in the first place, but there's plenty of people who view this as an overly restrictive setting, so it might not be the default. Does WS\_FTP Server fail in this regard?

Since connections to addresses detailed in PORT commands come from port 20 on the FTP server, this allows some measure of protection – most services should block connections from port 20. FTP servers are already recommended

SecurityFocus Bugtraq: Re: IPSwitch, Inc. WS\_FTP Server

(see RFC 2577) not to connect to ports lower than 1024.

- > *The PASV connection hijacking vulnerability allows a remote attacker*
- > *to intercept directory listings and file downloads from other users; file*
- > *uploads may also be spoofed. No authentication is necessary to execute*
- > *this attack. More information on this vulnerability can be found here:*
- > <http://www.kb.cert.org/vuls/id/2558>.

This is, contrary to the assertion at the web site listed above, a vulnerability in the `_client_`. There are several FTP clients that will send a PASV command followed immediately by a LIST, RETR, STOR, or whatever command, when they should be first connecting to the PASV port, and verifying that the connection was accepted before they send the command. As your example shows, if it is possible to guess the port that a server will be listening on, it's possible to make a connection to that port ahead of the client. A client that doesn't bother to consider this possibility (particularly since it's such a widely-known attack) is fundamentally flawed.

The best that a server can do against PASV hijacking is to improve the randomness of its choice of ports, and to close all connections other than the first received on the incoming port. It might also care to verify that the source address matches that of the client, but that, too, is somewhat a matter of taste. Again, does WS\_FTP Server fail to offer any of these options?

This is fundamentally a problem caused by a client requesting a transfer on a channel that it has not verified to be open. Stupid client behaviour does not make for a good report of flaws against the server.

For a brief time, we locked our server software down, and would close any socket that connected on the PASV port prior to our receiving a transfer command, but that turned out to have two problems:

1. It didn't fix the problem – faulty clients were only banned `_sometimes_`. More frequently, a functional client would be banned, when the command was received after the three-way handshake had been completed, but before we could get notice of it from the stack.
2. People pointed the finger at the wrong culprit – connectivity with one particularly well-known faulty client (okay, so it was Internet Explorer – big surprise) suffered, and people chose to drop our product rather than beg Microsoft to have pity on them.

SSL support in FTP stands as one good workaround for these issues. I'm working on another, which will eventually be put forward to the IETF FTPEXT Working Group.

Alun.

~~~~~

--

Texas Imperial Software | Try WFTPD, the Windows FTP Server. Find us at  
1602 Harvest Moon Place | <http://www.wftpd.com> or email [alun@texas.com](mailto:alun@texas.com)  
Cedar Park TX 78613-1419 | VISA/MC accepted. NT-based sites, be sure to

Re: IPSwitch, Inc. WS\_FTP Server

## SecurityFocus Bugtraq: Re: IPSwitch, Inc. WS\_FTP Server

Fax/Voice +1(512)258-9858 | [read details of WFTPD Pro for NT.](#)

---

- **Previous message:** Sym Security: "RE: DH team: Norton Antivirus Corporate Edition Privilege Escalation. <http://online.securityfocus.com/archive/1/296979/2002-10-22/2002-10-28/0>"
- **In reply to:** dev-null@no-id.com: "IPSwitch, Inc. WS\_FTP Server"
- **Next in thread:** 3APA3A: "Re[2]: IPSwitch, Inc. WS\_FTP Server"
- **Reply:** 3APA3A: "Re[2]: IPSwitch, Inc. WS\_FTP Server"
- **Messages sorted by:** [ date ] [ thread ] [ subject ] [ author ] [ attachment ]