

Re: ipfw or ipf w/stateful behavior

Source: <http://www.derkeiler.com/Mailing-Lists/FreeBSD-Security/2003-08/0011.html>

From: michael (michael_at_netmail.de)

Date: 08/03/03

To: Matthew Seaman <m.seaman@infracaninophile.co.uk>

Date: Sun, 03 Aug 2003 18:24:35 +0200 (CEST)

hello Matthew,

first thank you for these detailed report from the ftp-protokoll,
i think may know that you have to spend a lot of time, alot of koffee,
o lot of hard nights to learn this by working everyday.

Well, i know all these technics and way where the ftp-protokoll
goes. And also i remember "the good old spirit times" where nobody thinks
really over stateful firewalls and peer-to(o)-peer-Clients like
kaaza, gnutella, morpheus or icq-clients who use https/or http-port....

in honor of the good old times my eyes trop down one or two tears....

may i would not really use linux for my firewaal-box,
i would use freebsd with so less as possible third-party-software
like the ports. I use standart-ports on most systems i setted up
like rsync or webalizer and sqmlog, squid and thttp for reviewing the reports.

well, back to the essentials:

under linux can i load a kernelmodule for masquerading ftp-connections and
this allows me to close any port from outside except the ports for
Management or administration. these make the firewall secure enough.

May under FreeBSD it give no KLD_MODULE that solve the problem with ftp/or
irc. and i would not open any port unnecessary it would be used directly with an
time-out to close it up if no data flows trough the socket's.

The essential question from me is give it with ipf a solution to
solve this problem, the packet's must bee readable by ipf and ipf must held a
trackin-table for outgoing connections who also observe a little bit exclusive
any ftp-connection or any irc connection, with corresponding the src- and dst-ip.

thanks you very much

best regards and have a good and funny time

FreeBSD–Security: Re: ipfw or ipf w/stateful behavior

michael

Quoting Matthew Seaman <m.seaman@infracaninophile.co.uk>:

> *On Sun, Aug 03, 2003 at 02:41:32PM +0200, michael wrote:*
>
> > *Now i have made all rules with the setup/established or keep–state*
> *flags*
> > *(ipfw) and my ftp–connections are not really stateful. I think*
> > *that these behavior is also so by irc–chat.*
> >
> > *Now i wont to know, how must i do to become also an stateful behavior*
> > *for these services, w/o to open the high–ports from the firewall,*
> > *then at the last time i become over and over with portscans from*
> *outside,*
> > *and i think this is an security reason.*
> >
> > *i don't really want to open the high–ports on my box.*
>
> *I take it you're trying to access a remote FTP server, not that you're*
> *hosting a FTP server at your site? Securing things for an FTP client*
> *is rather easier than for an FTP server, especially if you've got a*
> *NAT gateway in between.*
>
> *The problem with FTP is that it is one of the oldest designs of any of*
> *the commonly used networking protocols, and it suffers from a number*
> *of flaws not found in more modern protocols like HTTP. In the days*
> *when it was first designed and implemented, the concept of*
> *automatically using a packet filtering firewall to protect servers and*
> *clients really hadn't yet achieved any real credence. Consequently*
> *the designers felt free to do things like require two connections*
> *between the client and the server: one channel for data and the other*
> *for control messages. See the first part of*
> *<http://www.faqs.org/rfcs/rfc959.html> (1985) for a potted history of*
> *the protocol.*
>
> *Traditionally the way an FTP session has been set up is:*
>
> *i) Client connects to port 21 (ftp control) on the server. This*
> *establishes the control channel, which is used throughout the*
> *session. The client side port is just an arbitrary*
> *high–numbered port as used for any outgoing connection.*
>
> *ii) Client can issue various FTP protocol commands however, as*
> *soon as a command is sent that requires data to be returned*
> *(eg. asking for a directory listing) or if a file has to be*
> *transferred in either direction, then the client sends a PORT*
> *command to the server, which tells the server to open up a*
> *data connection typically from port 20 (ftp–data) on the*
> *server end to the given port number on the client. This can*
> *happen several times during an FTP session, if more than one*

Re: ipfw or ipf w/stateful behavior

FreeBSD–Security: Re: ipfw or ipf w/stateful behavior

- > *file is transferred.*
- >
- > *Under FreeBSD the client side port number will be bounded by*
- > *the port numbers given in the net.inet.ip.portrange.hifirst*
- > *and net.inet.ip.portrange.hilast sysctls, which will usually*
- > *be something like 49152 -- 65535, but see the 'restrict'*
- > *command in ftp(1).*
- >
- > *Now, this is somewhat horrifying to the modern client–side network*
- > *administrators: either you've got to install a protocol aware*
- > *firewall, that can detect the outgoing PORT command (with all the*
- > *pitfalls that entails) and poke just the right hole in the firewall to*
- > *allow the incoming data connection or you've got to bite the bullet*
- > *and let external systems make arbitrary incoming connections to the*
- > *high port range of your systems. As far as I know, there isn't a FTP*
- > *protocol aware firewall implementation freely available for FreeBSD*
- > *(although Checkpoint FW–1 is a commercial product that can do that so*
- > *of thing, but the closest it gets to running on FreeBSD is when it's*
- > *sold as part of a Nokia firewall appliance: those have an OS called*
- > *IPSO which is apparently based on FreeBSD 2.x or 3.x)*
- >
- > *Instead, and pretty much standard nowadays, an FTP client will use*
- > *passive–mode FTP. All web browsers, when told to retrieve a ftp://*
- > *URL will automatically use passive ftp. Under FreeBSD, you can set*
- > *FTP_PASSIVE_MODE in the environment and the bundled ftp(1) and*
- > *fetch(1) FTP clients will then assume passive mode, or you can use the*
- > *ftp(1) 'passive' command within an FTP session to toggle passive mode*
- > *on or off.*
- >
- > *In this case the sequence of events is:*
- >
- > *i) Client connects to port 21 on the server, as before.*
- >
- > *ii) Now, when it is necessary to open up the data channel, the*
- > *client sends the server a PASV command, to which the server*
- > *replies with a suitable port number that the client can open a*
- > *connection to. This time it's the server that uses a port in*
- > *the 49152..65535 range (although see the documentation of the*
- > *–U option in ftpd(8) for ways to modify that, and the client*
- > *end generally uses some arbitrary port as for any outgoing*
- > *connection.*
- >
- > *Here both connections are opened by the client onto the server, which*
- > *is much more friendly to the client side firewall. You can just write*
- > *a rule (stateful or not, as you choose) to permit outgoing connections*
- > *-- either to the high range ports, or more generally to any port:*
- >
- > *ipfw add allow tcp from \${myip} to any 49152–65535 keep–state out*
- > *xmit \${oif}*
- >
- > *(omit the 49152–65535 part if you want to allow all outgoing*

FreeBSD–Security: Re: ipfw or ipf w/stateful behavior

> connections. $\{myip\}$ is your local IP address range, and $\{oif\}$ is
> the outward facing ethernet interface on your firewall: eg. fxp0, rl1)
>
> Note that people that run FTP servers would generally prefer to use
> the original PORT style, rather than PASV so that they in their turn
> could write nice tight firewall rules. However, as that would prevent
> most clients accessing their FTP archives they pretty well have to
> provide PASV support.
>
> > give it an chance by using ipf and not ipfw??
>
> Either ipfw(8) or ipf(8) should be able to do the job for you:
> functionally the two are quite similar but the configuration syntax is
> fairly different.
>
> > i have read the documentations, and i have no hint found
> > that solve this problem, my i have seen that in first time
> > ipf is mutch more complex to configure and has more pitfalls
> > to make mistakes, with the ip packet description language.
>
> Stick with what suits you the best.
>
> Cheers,
>
> Matthew
>
> --
> Dr Matthew J Seaman MA, D.Phil. 26 The Paddocks
> Savill Way
> PGP: <http://www.infracaninophile.co.uk/pgpkey> Marlow
> Tel: +44 1628 476614 Bucks., SL7 1TH
> UK
>

freebsd–security@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd–security>

To unsubscribe, send any mail to "freebsd–security–unsubscribe@freebsd.org"