Using the Intrusion Prevention System in MultiNet V5.3

Jeremy Begg
VSM Software Services Pty Ltd
March 2009

Introduction

One of the interesting new features in MultiNet V5.3 is a security mechanism called the "Intrusion Prevention System". With IPS you can configure MultiNet to detect intrusion attempts and then block further access from the remote system that is the source of the attack.

IPS is documented in Chapter 32 of the MultiNet V5.3 Installation & Administration Guide.

Why use IPS?

Here at VSM Software Services we run an AlphaServer DS20E to provide services to customers on the Internet including DNS, webserving and email. We also allow incoming FTP and SSH access for server management and customer website maintenance.

There's no doubt that the Internet has become a hostile place, even for servers running OpenVMS. In operating these services we regularly see VMS breakin events logged by MultiNet's SSH and FTP servers and by PMDF's POP3, IMAP and SMTP servers. We also see suspicious DNS update attempts but (fortunately for us!) not much in the way of Denial-of-Service attacks (*i.e.* attempts to flood the server with so much traffic that it can't maintain acceptable responsiveness).

To date we've used a variety of methods to control this activity, all of them relying on manual observation of an attack and manual updating of a configuration file. For example,

- PORT_ACCESS mapping rules in PMDF to block access from persistently annoying systems
- Configuring DNS to refuse recursive lookups from outside the local network
- Manually updating MULTINET:FILTER-SE0.DAT to block all IP traffic from specified hosts.

MultiNet IPS uses information provided by network services to detect suspicious activity and then dynamically updates the MultiNet packet filter to disrupt that activity. (By "network services" I mean server processes on the local system which provide TCP/IP application services such as SSH, POP3 or HTTP.)

We already had a lot of experience with MultiNet's packet filter mechanism so the ability to have it automatically updated by IPS had a lot of appeal.

Configuring IPS

IPS is installed as part of the standard MultiNet installation but until its configuration files are set up it doesn't do anything. These files are described in detail in the

MultiNet Installation & Administration Guide (Chapter 32) but briefly the procedure is as follows:

1. Copy MULTINET:FILTER_SERVER_CONFIG.TEMPLATE to MULTINET:FILTER_SERVER_CONFIG.TXT.
2. Edit MULTINET:FILTER_SERVER_CONFIG.TXT and set desired configuration options. In particular, specify the service-specific configuration files which will be loaded using INCLUDE statements.
3. Set up each of the service-specific configuration files specified in step 2. For example, copy MULTINET:SSH_FILTER_CONFIG.TEMPLATE to MULTINET:SSH_FILTER_CONFIG.TXT.
4. Edit each of the service-specific configuration files to specify the criteria for each service, such as the template packet filter rule and the trigger for activating the packet filter.
5. When all configuration files have been prepared, issue the command
   `$ MULTINET SET/IPS/RELOAD`
   to activate them.

Services Protected by IPS

The MultiNet installation kit comes with templates for protecting the most common MultiNet services including FTP, IMAP, POP3, REXEC, RLOGIN, RSHELL, SMTP, SNMP, SSH and TELNET.

If your system is also running PMDF V6.4 you can use IPS to protect the PMDF IMAP, POP3 and SMTP servers. The configuration files for these are shipped in the PMDF_TABLE: directory.

The service-specific configuration file

There is a configuration file for each service. Reasonable defaults are provided in the .TEMPLATE files but some changes are required:

`destination_address` This must match the IP address (in CIDR format) of the interface to be monitored for intrusion activity. For example,

```
destination_address 150.101.13.12/27
```

This specifies that the interface with IP address 150.101.13.12 is to be monitored. The subnet portion of the address (/27 in this case) is required but ignored.

`exclude_address` This specifies one or more remote IP addresses which are to be ignored, *i.e.* the packet filter will never block those addresses. The template files come with this configured to block a commonly-used local address (192.168.0.10/24) but sites may wish to remove or change it.

Each template file also specifies a prototype packet filter entry which looks a little odd at first glance:

```
proto_filter "deny tcp 192.168.0.100/24 192.168.0.1/24 log"
```

The two IP address ranges in CIDR format will be automatically replaced by the actual source and destination addresses when the rule is activated by MultiNet IPS. This rule only needs to be changed if you wish to change the action, *e.g.* from “deny” to “drop”.

Monitoring the IPS

MultiNet Intrusion Prevention System generates a wealth of evidence for its effectiveness including assorted MultiNet log files, OPCOM, SNMP and the OpenVMS Security Audit logs.

There are several log files created by MultiNet IPS:

- MULTINET:FILTER_SERVER.OUT is the primary log file for the filter server process.
- MULTINET:FILTER_SERVER_HOURLY_LOG.*yyyymmdd* is a "day file" containing a summary of filter actions each hour during the day. A new file is created every day at 1am.

Here is an extract from the FILTER_SERVER.OUT file on one of our systems:

```
FILTER_SERVER V1.0.0

20-MAR-2009 12:14:07.28 - Using configuration file
MULTINET_ROOT:[MULTINET]FILTER_SERVER_CONFIG.TXT;
20-MAR-2009 12:14:07.30 - Processing include file "multinet:ssh_filter_config.txt"
20-MAR-2009 12:14:07.30 - Using configuration file
MULTINET_ROOT:[MULTINET]SSH_FILTER_CONFIG.TXT;
20-MAR-2009 20:37:03.46 - Event message received
20-MAR-2009 20:37:03.46 -     Component: SSH
20-MAR-2009 20:37:03.46 -     Rule     : SSH_BOGUS_ID
20-MAR-2009 20:37:03.46 -     Time     : 20-MAR-2009 20:37:03.46
20-MAR-2009 20:37:03.47 -     Src Port : 54232
20-MAR-2009 20:37:03.47 -     Src Addr : 68.54.152.69
20-MAR-2009 20:37:03.47 -     Dst Addr : 150.101.13.12
20-MAR-2009 20:37:03.47 -     Process  : SSHD Master
20-MAR-2009 20:37:03.47 -     PID      : 208000AD
20-MAR-2009 21:12:01.31 - Event message received
20-MAR-2009 21:12:01.31 -     Component: SSH
20-MAR-2009 21:12:01.31 -     Rule     : SSH_INVALIDUSER
20-MAR-2009 21:12:01.31 -     Time     : 20-MAR-2009 21:12:01.31
20-MAR-2009 21:12:01.32 -     Src Port : 55839
20-MAR-2009 21:12:01.32 -     Src Addr : 68.54.152.69
20-MAR-2009 21:12:01.32 -     Dst Addr : 150.101.13.12
20-MAR-2009 21:12:01.32 -     Process  : SSHD 0000
20-MAR-2009 21:12:01.32 -     PID      : 20800B83
20-MAR-2009 21:12:06.49 - Event message received
20-MAR-2009 21:12:06.49 -     Component: SSH
20-MAR-2009 21:12:06.49 -     Rule     : SSH_INVALIDUSER
20-MAR-2009 21:12:06.49 -     Time     : 20-MAR-2009 21:12:06.49
20-MAR-2009 21:12:06.50 -     Src Port : 55991
20-MAR-2009 21:12:06.50 -     Src Addr : 68.54.152.69
20-MAR-2009 21:12:06.50 -     Dst Addr : 150.101.13.12
20-MAR-2009 21:12:06.50 -     Process  : SSHD 0001
20-MAR-2009 21:12:06.50 -     PID      : 20800C04
.
... Event messages removed for brevity ...
.
20-MAR-2009 21:12:47.13 - Event message received
20-MAR-2009 21:12:47.13 -     Component: SSH
20-MAR-2009 21:12:47.13 -     Rule     : SSH_INVALIDUSER
20-MAR-2009 21:12:47.13 -     Time     : 20-MAR-2009 21:12:47.12
20-MAR-2009 21:12:47.13 -     Src Port : 50039
20-MAR-2009 21:12:47.13 -     Src Addr : 68.54.152.69
20-MAR-2009 21:12:47.14 -     Dst Addr : 150.101.13.12
20-MAR-2009 21:12:47.14 -     Process  : SSHD 0009
20-MAR-2009 21:12:47.14 -     PID      : 20800C20
20-MAR-2009 21:12:47.14 - Creating a filter for component ssh rule ssh_invaliduser
20-MAR-2009 21:12:47.14 -                   src address   =  68.54.152.69/32
20-MAR-2009 21:12:47.14 -                   dst address   =  150.101.13.12/27
20-MAR-2009 21:12:47.14 -                   interface     =  se0
20-MAR-2009 21:12:47.14 -                   filter expires   20-MAR-2009 21:17:47.14
21-MAR-2009 00:00:00.50 - Performing daily maintenance
```

The extract above shows that the filter server started at 20-MAR-2009 12:14:07.28 and loaded a single service-specific configuration file (for SSH). At 20:37:03.46 the SSH server reported suspicious activity but this was not followed by any other such activity within the specified timeout (5 minutes by default) and so was ignored. Then at 21:12:01 the SSH server reported more suspicious activity and this time the remote system (68.54.152.69) persisted in its breakin attempts. After ten such reports in the space of under a minute the IPS created a packet filter to block the remote system.

The FILTER_SERVER_HOURLY_LOG files contain hourly snapshots of IPS activity. For example the time period corresponding to the extract above looks like this:

```
Filter server hourly snapshot for hour 21 of 03/20/2009

Component  ssh

    Rule ssh_bogus_id
        number of hits:      0
        destination address: 150.101.13.12/27

        Address 68.54.152.69/32
            number of still-queued events:   0
            number of all events:            0
            number of filters created:       0
            Address entry to be deleted:     21-MAR-2009 00:42:03.46

    Rule ssh_authfailed
        number of hits:      0
        destination address: 150.101.13.12/27

    Rule ssh_userauth
        number of hits:      0
        destination address: 150.101.13.12/27

    Rule ssh_invaliduser
        number of hits:      10
        destination address: 150.101.13.12/27

        Address 68.54.152.69/32
            number of still-queued events:   0
            number of all events:            10
            number of filters created:       1
            Address entry to be deleted:     21-MAR-2009 01:12:47.19
```

In addition to the log files you can use the regular MultiNet interface commands to see what packet filters are in place at any given moment:

```
$ mu show/int se0/filter
Device se0: flags=8863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,D2>
            VMS Device = EWA0
            IP Address = 150.101.13.12
            No common links defined

MultiNet Packet Filter List for se0:

Logging is disabled

                       Source Address / Port
Action  Proto   Hits   Destination Address / Port
------  -----  -----  --------------------------------------------------
drop      tcp     29  213.174.151.17/32
                      150.101.13.0/27
                      FLTSVR,LOG
$
```

(Note that the output above was generated some days after the events shown in the log file extracts.)

Closing Remarks

We're still in the early days of using IPS here but it's already proven to be effective at limiting intrusion activity. Here are some thoughts about it which you might find useful.

- In setting up the SSH filters for IPS we chose to change the default *proto_filter* rule from "deny" to "drop". In our experience with SSH-based attacks the remote systems keep sending packets to the server even though they're getting "administratively denied" responses. Changing the rule to "drop" causes nothing whatsoever to be sent back to the remote systems, and they seem to stop trying much sooner.
- The default trigger for these events is 10 occurrences inside 5 minutes. At our site where most SSH users are what you might call "sophisticated" we could probably tighten this to (say) three login failures inside 5 minutes. On the other hand if we had a large number of users the probability that any given user would enter the wrong password would be somewhat higher, and "3 in 5" might be too restrictive.
- Once we get PMDF upgraded on our primary server we'll look at implementing IPS for PMDF's IMAP, POP3 and SMTP servers. All of them are popular targets for password-hunting netbots.

I hope you've found the information in this white paper to be useful and that it encourages to make use of MultiNet's Intrusion Prevention System to good effect.