Maximum Netfilter

Michael Rash
Founder, http://www.cipherdyne.org/
mbr@cipherdyne.org

OSCON
Portland, Oregon
07/26/2006

Agenda

- Netfilter and Enterprise functionality
- Defense In Depth
- Attack detection and response
- Protocols and the Netfilter logging format
- Snort rule translation with fwsnort
- Summarization, reporting, and response with psad
- Single Packet Authorization with fwknop

Enterprise Functionality

- Granular filtering capability (including state tracking)
- NAT
- Application layer inspection
- GUI interface support with Fwbuilder
- Comprehensive Logging
- Performance
- Active Development
- Low Cost

Defense In Depth

- Application layer inspection makes it possible to supplement intrusion detection systems
- Netfilter is inline
- Active response capabilities (REJECT and DROP targets)
- Kernel level filtering implies default DROP rules severely limit stack access

Attack Detection

- Snort signature rule set is the industry standard
- Snort-2.3.3 rules are released under the GPL
- Rules from http://www.bleedingsnort.com are released under a BSD-style license
- Over 3,000 signatures; 95% of which require application layer tests

Example Snort Rules

Transport layer flags (no application layer inspection):

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"SCAN XMAS"; **flow**:stateless; **flags**:SRAFPU,12; reference:arachnids,144; classtype:attempted-recon; sid:625; rev:7;)

Application layer inspection:

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-ATTACKS /usr/bin/gcc command attempt"; **flow**:to_server,established; **content**:"/usr/bin/gcc"; **nocase**; classtype:web-application-attack; sid:1341; rev:5;)

Can Netfilter Emulate Snort Rule Detection?

- Nearly every interesting field in the network and transport layer headers is logged
  - source and destination Ips/Networks
  - source and destination port numbers
  - protocol, sameip, id, ttl, tos, ipopts, itype, icode, icmp_seq, icmp_id
  - flags, flow, window
  - ack, seq (requires --log-tcp-sequence)

Netfilter Logs – ICMP Packet

Jul 25 15:49:06 netfilter kernel: **IN**=eth0 **OUT**=
MAC=00:13:d3:38:b6:e4:00:01:5c:22:9b:c2:08:
00 **SRC**=68.50.49.x **DST**=68.50.139.x LEN=84
TOS=0x00 PREC=0x00 TTL=64 ID=32128
PROTO=ICMP **TYPE**=0 **CODE**=0 **ID**=30739
SEQ=1

Netfilter Logs – UDP Packet

Jul 25 16:07:10 netfilter kernel: IN=eth0 OUT=
MAC=00:13:d3:38:b6:e4:00:01:5c:22:9b:c2:08:
00 **SRC**=68.50.49.x **DST**=68.50.139.x **LEN**=33
TOS=0x00 PREC=0x00 TTL=64 ID=1247 DF
PROTO=UDP **SPT**=32774 **DPT**=53 **LEN**=13

Netfilter Logs – TCP Packet

Jun 17 23:57:20 netfilter kernel: DROP IN=eth0 OUT=
MAC=00:13:d3:38:b6:e4:00:01:5c:22:9b:c2:08:00
SRC=68.50.49.x **DST**=68.50.139.x LEN=48
TOS=0x00 PREC=0x20 **TTL**=122 **ID**=45072 DF
PROTO=TCP SPT=1993 **DPT**=80 WINDOW=65535
RES=0x00 **SYN** URGP=0 OPT (**020405B401010402**)

Can Netfilter Emulate Snort Rule Detection? (con'td)

- The string match extension (now available in the stock kernel as of 2.6.14)
  - Snort **content** field
- End result is that approximately 50% of all Snort rules can be translated into equivalent Netfilter rules
- Fwsnort automates the translation process

Fwsnort Rule Translation

```
# fwsnort --snort-sid=1341
[+] Parsing Snort rules files...
[+] Found sid: 1341 in web-attacks.rules
    Successful translation.

[+] Logfile:          /var/log/fwsnort.log
[+] Iptables script:
/etc/fwsnort/fwsnort.sh
```

Translated Snort Rule

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-ATTACKS /usr/bin/gcc command attempt"; **flow:to_server,established**; content:"/**usr**/**bin**/**gcc**"; nocase; classtype:web-application-attack; **sid:1341**; rev:5;)

Becomes:

iptables -A **FWSNORT_INPUT_ESTAB** -p tcp --dport 80 -m string --string "/**usr**/**bin**/**gcc**" --algo bm -j LOG --log-prefix "**SID1341 ESTABLISHED** "

Translated Snort Rule (cont'd)

Optionally (with “**fwsnort --snort-sid 1341 --ipt-drop**”):

iptables -A FWSNORT_INPUT_ESTAB -p tcp --dport 80 -m string --string "/usr/bin/gcc" --algo bm -j LOG --log-prefix "**DRP** SID1341 ESTABLISHED "

iptables -A FWSNORT_INPUT_ESTAB -p tcp --dport 80 -m string --string "/usr/bin/gcc" --algo bm -j **DROP**

Lost in Translation

- Why only 50% translation rate?
  - Unsupported Snort options
    - pcre
    - flowbits
    - byte_test
    - byte_jump
    - asn1
    - content-list
    - distance
    - within

Reporting?

- There is a difference between filtering and logging in Netfilter.
- Netfilter log prefixes are limited to 30 characters, so logging application layer data is not practical; Snort rule ID values fit however
- Logging vs. filtering issues aside, there still needs to be a mechanism for effective alerting

Psad

- Psad email and syslog alerting
- Scan detection (SYN, FIN, XMAS, NULL, UDP)
- Reporting for Fwsnort “SIDnnn” messages (includes class type and reference information)
- Passive OS fingerprinting
- DShield reporting
- Persistent timeout-based blocking rules

Single Packet Authentication

- Developing secure software is hard
- Cisco IOS Firewall Authentication Proxy Buffer Overflow Vulnerability
- IPSec ESP Information Leak Vulnerability
- Check Point FW-1 Authentication Vulnerability
- OpenSSH GSSAPI Credential Disclosure Vulnerability

Cleartext IDS Over Encrypted Protocols

- EXPLOIT gobbles SSH exploit attempt
- EXPLOIT ssh CRC32 overflow NOOP
- EXPLOIT ssh CRC32 overflow filler

perl -e 'print “A”x1000' | nc <target> 80

(.)\1{500}

Why Another Authentication Method?

- Strong crypto NOT enough
- Nmap has solved the mapping problem
- Exiting methods assume TCP/IP stack access; they will find you!

fwknop and SPA

- Netfilter Default DROP stance for protected services
- Dynamic reconfiguration of rule set upon receiving a valid SPA packet
- SPA is encrypted (both symmetric and asymmetric algorithms can be used)
- SPA is non-replayable
- Any IP protocol can be used

Live Demonstration...

Questions?

mbr@cipherdyne.org

http://www.cipherdyne.org/