Netfilter and Encrypted, Non-Replayable, Spoofable, Single Packet Remote Authorization

Michael Rash
Enterasys Networks, Inc.

ToorCon

09/17/2005

http://www.cipherdyne.org
Agenda

- Vulnerabilities and trends
- Target enumeration
- Single Packet Authorization (SPA)
- Fwknop design and implementation
- Live demo
- Future development
Vulnerability Goulash

- IPsec ESP Information Leak Vulnerability
- Cisco IOS Firewall Authentication Proxy Buffer Overflow Vulnerability
- Check Point FW-1 Authentication Vulnerability
- OpenSSSH 3.x scp Input Validation Vulnerability
- OpenSSSH 3.x CRC32 Overflow
Potential Compromise vs. Convenience

• 50 new vulnerabilities per day
  – http://www.idefense.com

• Authorization methods and strong encryption is not enough

• VPN access is essential!
# nmap -P0 -p T:22,256 -sS -sV 192.168.10.1

Starting nmap 3.81 ( http://www.insecure.org/nmap/ ) at 2005-08-04 22:06 EDT

Interesting ports on 192.168.10.1:

<table>
<thead>
<tr>
<th>PORT</th>
<th>STATE</th>
<th>SERVICE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>22/tcp</td>
<td>open</td>
<td>ssh</td>
<td>OpenSSH 3.9p1 (protocol 2.0)</td>
</tr>
<tr>
<td>256/tcp</td>
<td>closed</td>
<td>FW1-secureremote</td>
<td></td>
</tr>
</tbody>
</table>

Nmap finished: 1 IP address (1 host up) scanned in 0.139 seconds
Target Enumeration (cont'd)

# host www.yahoo.com
www.yahoo.akadns.net has address 216.109.117.206

# whois 216.109.117.206 | grep CIDR
CIDR: 216.109.112.0/20

# nmap -P0 -p T:22,256 -sS -sV -T Aggressive
216.109.112.0/20
Default Drop

# iptables -I INPUT 1 -p tcp --dport 22 -j DROP
Single Packet Authorization (SPA)

- Use packet filters to minimize execution paths
- Passive monitoring of packet data (all hail libpcap!)
- No traditional “server”
Single Packet Authorization (cont'd)

- Asymmetric or symmetric encryption
- Authorization packets can be spoofed
- Any IP protocol can be used
- Up to minimum MTU for data transmission
- Works across NAT
Single Packet Authorization vs. Port Knocking

- Both techniques use packet filters
- Much more data can be sent with SPA
- Protocols without a notion of a “port” can be used
- No port sequences to bust
- Replay attacks easily thwarted
- More difficult to detect (nothing to mistakenly identify as a port scan)
Fwknop

- pcap, file_pcap, Netfilter pcap writer
- Data collection methods
- Rijndael symmetric block cipher
- Packets prepended with 16 bytes of random data
- Supports multiple remote users
- Message integrity verified via internal MD5 sum
- Integrates with NAT
Fwknop (cont'd)

- Built-in spoofing capability (Net::RawIP)
- Supports TCP, UDP, ICMP (default UDP/62201)
- Message replays stopped via MD5 sum cache
- Integrates with Netfilter policy via custom chains
- Supports access and command modes
Fwknop (con'td)

- Client runs on Linux, OS X, and FreeBSD (others?)
- Installer preserves configs across upgrades
- Server supports syslog messages and email alerts

Sep 13 21:15:58 orthanc fwknop: received valid encrypted packet from: 192.168.10.2, remote user: mbr

Sep 13 21:15:58 orthanc fwknop: adding FWKNOP_INPUT ACCEPT rule for 192.168.10.2 -> tcp/22 (10 seconds)

Sep 13 21:16:09 orthanc fwknop: removed iptables FWKNOP_INPUT ACCEPT rule for 192.168.10.2 to tcp/22, 10 second timeout exceeded
Deployment Architecture

Client

Fwknop Sniffer

Attacker2

Dummy Target IP

Attacker3

Attacker1

FW
Fwknop Usage

- Standard /etc init script for server mode
- Debug modes for both client and server

  - fwknop -A tcp/22,tcp/256 --Spoof-src www.yahoo.com -a <MY IP> -k <target>
  - fwknop --Server-cmd "ping -c 3 www.yahoo.com" -s -k <target>
Packet Format

Random data: 7808936091987532
Username: mbr
Timestamp: 1123247144
Version: 0.9.1
Action: 1 (access mode)
Access: 0.0.0.0,tcp/22
MD5 sum: y6tuSWoS+py7ppsESNR78A
Encrypted Packets

udp/62201 (128 bytes):

Hul72UvwLqLqxiQLfTi7nXyjqIr37s8R9/JrYGcaP9PI4ADNK9pqeFghA20pXHwpQf/TAbxt1L+GSwAkJBSP0USBRm6IK87+xBAVRpb9UNJ8HUw3DsRTXpcYXtqrPQP

ISTLpc2VMs2jGOJsJOAwIWxKChKUOMS88PteezX6u7TCsd7KVgzOIvjPRuSckjP/tbInEeMUK+53tKfvifNIX5vODinG5Cyi96XZThF2NO53dWN1dzQMv3dwPfbZdCab
Netfilter Integration

- Compatible with existing policy
- Most effective with connection tracking enabled
- Custom fwknop chains (FWKNOP_INPUT)
- Optional data collection via ULOG target
Example Netfilter Policy

Chain INPUT (policy ACCEPT)

FWKNOP_INPUT  all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT        all -- 0.0.0.0/0 0.0.0.0/0  state RELATED,ESTABLISHED
ACCEPT        tcp -- 192.168.10.3 0.0.0.0/0 tcp dpt:80
ULOG          udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:62201 ULOG
copy_range 0 nlgroup 1 prefix `FWKNOP' queue_threshold 1

Chain FWKNOP_INPUT (1 references)

ACCEPT        tcp -- * * 192.168.10.2 0.0.0.0/0 tcp dpt:22
Fwknop Server Config

- **fwknop.conf**
  - Defines data collection mode, email alert address(es), and file paths

- **access.conf**
  - Defines access controls for fwknop clients
fwknop.conf

EMAIL_ADDRESSES          mbr@cipherdyne.org;
AUTH_MODE                ULOG_PCAP;
PCAP_INTF                eth1;
ENABLE_PCAP_PROMISC      Y;
PCAP_FILTER              udp port 62201;
PCAP_PKT_FILE            /var/log/ulogd.pcap;
ENABLE_MD5_PERSISTENCE   Y;
SOURCE: ANY;
DATA_COLLECT_MODE: ULOG_PCAP;
OPEN_PORTS: tcp/22;
PERMIT_CLIENT_PORTS: Y;
#ENABLE_CMD_EXEC: Y;
#CMD_REGEX: echo\s+\S+\s*>>;
KEY: <encryptkey>;
FW_ACCESS_TIMEOUT: 10;
REQUIRE_USERNAME: mbr;
IDS Alert Reduction

- Most IDS's are stateful
- Sessions can only be established after authorization
Live Demo...
Disadvantages

- Additional key management
- Some services not readily compatible
- Session “piggy backing”
- Adds extra layer and associated time delay
- Authorization packets not transferred over reliable communication mechanism
Future Development

• Integration with PGP/GPG key rings
• Add support for existing authentication infrastructure (LDAP, Kerberos, Radius, etc.)
• Client integration (SSH, Web browsers)
• GUI development
• Potential kernel stack extensions (NDIS driver on Windows, IP stack patch for Linux)
Questions?

http://www.cipherdyne.org/fwknop/

mbr@cipherdyne.org