Zero-Day Attack Prevention via Single Packet Authorization

Michael Rash
Security Architect
Enterasys Networks, Inc.

http://www.cipherdyne.org/

2007.06.05
Techno Security

Agenda

- Passive authorization technologies (Port Knocking and Single Packet Authorization)
- Security Through Obscurity? (No!)
- The fwknop implementation of SPA
- New fwknop release: version 1.8
- Live demonstration

Where it all Started...

- Firewalls / Router ACL's – Packet filtering based on IP addresses, protocols, and services
- Port knocking
- Encrypted port knocking
- Single Packet Authorization (SPA) – Next generation passive authentication/authorization

Passive Authorization

- Prove you are a friend before you can connect to a service – try to minimize code paths available to an attacker
- Target enumeration is just too easy with Nmap
- Zero-day vulnerabilities will continue exist in server software

Recent OpenSSH Vulnerabilities

- Just search through http://www.securityfocus.com/bid/

  2007-05-08: PAM Authentication Remote Information Disclosure Vulnerability

  2007-04-10: Duplicated Block Remote Denial of Service Vulnerability

  2007-03-15: GSSAPI Credential Disclosure Vulnerability

  2007-03-14: GSSAPI Authentication Abort Information Disclosure Weakness

  2007-02-22: Enabled PAM Delay Information Disclosure Vulnerability

  2007-02-14: Existing Password Remote Information Disclosure Weakness

  - Can do this with any software vendor, not just OpenSSH (although OpenSSH is one of the most interesting examples)

Target Enumeration

[scanner]# host www.yahoo.com

www.yahoo.akadns.net has address 216.109.117.206

[scanner]# whois 216.109.117.206 | grep CIDR

CIDR: 216.109.112.0/20

[scanner]# nmap -P0 -p T:22 -sS -sV -T Aggressive 216.109.112.0/20

Snort IDS Inspection of Encrypted Protocols

- EXPLOIT ssh CRC32 overflow NOOP
- EXPLOIT ssh CRC32 overflow filler
- EXPLOIT gobbles SSH exploit attempt
- WEB-MISC SSLv3 invalid timestamp attempt
- EXPLOIT SSLv2 Client_Hello with pad Challenge Length overflow attempt

- Encryption does NOT imply an application is secure - some vulnerabilities may be accessible before encryption even becomes involved

Port Knocking

- Basic strategy: default-drop packet filter is reconfigured to allow temporary access to protected service after a specific sequence of connection attempts is passively monitored
- Example: connecting to the following ports results in temporary SSH access through the firewall
  - tcp/65510
  - udp/1022
  - udp/13100
  - tcp/15799

    ==> Access is granted to SSHD

Single Packet Authorization

- Firewall reconfigured after monitoring the following example packets via libpcap over UDP port 62201:

Hul72UvwLqLqxiQLfTi7nXyjqIr37s8R9/JrYGcaP9PI4ADNK9pqeFg
hA20pXHwdpQf/TAbxt1L+GSwAkJBSP0USBRm6IK87+xBaVRpb9
UNJ8HUw3DsRTXpcYXtqrPQP

==> Access is granted to SSHD

Passive Authorization is not STO

- Heated debate – BUT security *via* obscurity is bad, security *with* obscurity can help
- People don't say “I can't implement XYZ security measure because it makes my service more obscure”
- security-basics mailing list thread
  - http://dmiessler.com/study/security_and_obscurity/
  - http://seclists.org/basics/2007/Apr/0015.html

Passive Authorization is not STO (cont'd)

"...If I take a letter, lock it in a safe, hide the safe somewhere in New York, and then tell you to read the letter, that is not security. That's obscurity. On the other hand, if I take a letter, and then give you the safe along with the design specifications of the safe and a hundred identical safes with their combinations so that you and the world's best safe-crackers can study the locking mechanism – and you still can't open the safe – that's security..."

--Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*

Passive Authorization is not STO (con'td)

- Every port knocking and SPA implementation is open source – the complete specification (including all cipher implementations) is open for anyone to evaluate and critique
- SPA is about *concealment* – this is no more STO than passwords or encryption keys. People don't say “I have to give up my encryption keys and passwords now because otherwise my security suffers from STO”.
- Sebastien Jeanquier's M.S. Thesis “An Analysis of Port Knocking and Single Packet Authorization”
  - http://web.mac.com/s.j/

Port Knocking Projects

- Port knocking projects tracked at: http://www.portknocking.org/
- Nearly 30 projects at last count
- Martin Krzywinski, "*Port Knocking: Network Authentication Across Closed Ports*", SysAdmin Magazine, 2003

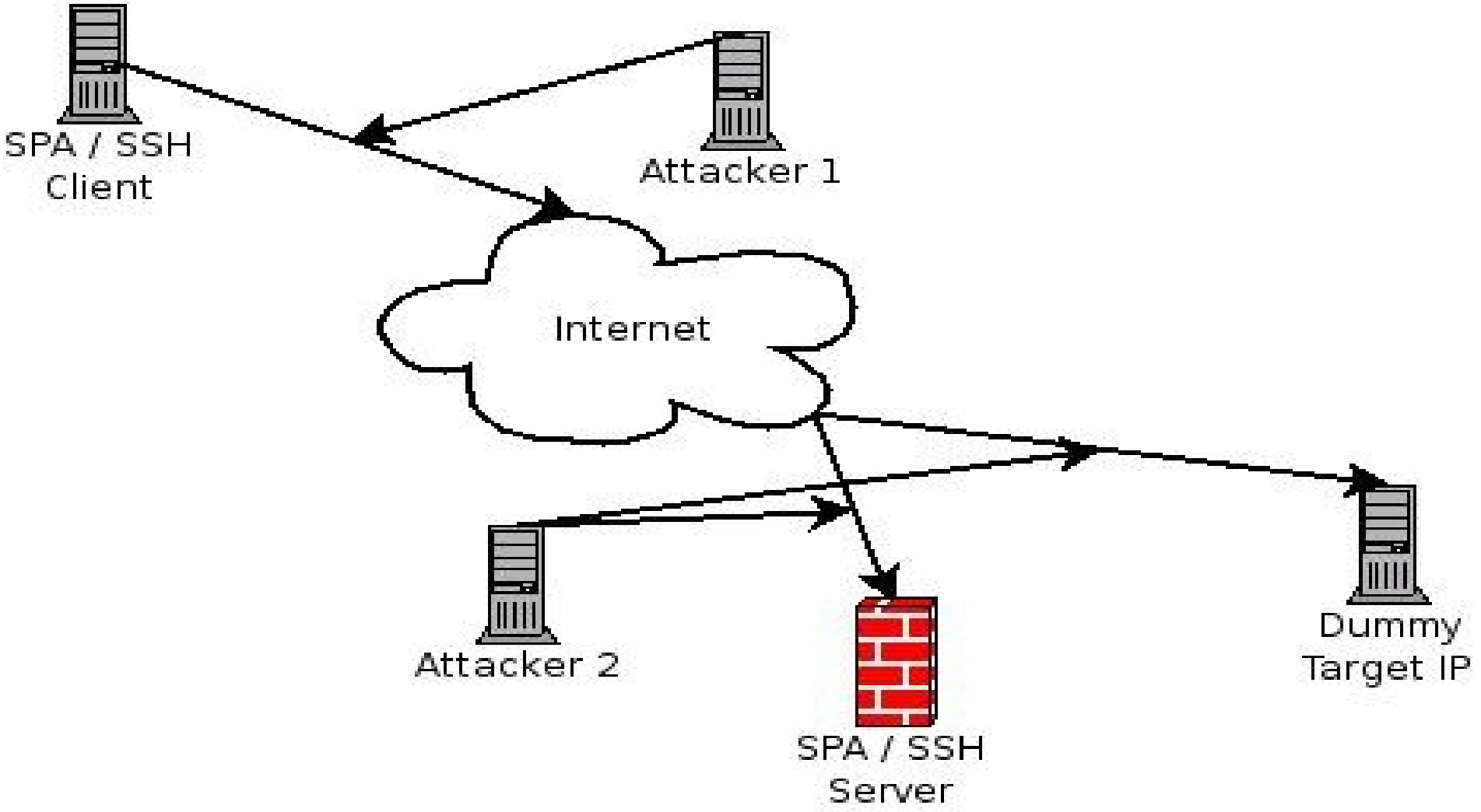
Port Knocking Protocol Weaknesses

- Knock sequences easily busted by a spoofed packet to a duplicate port – DoS attack against the client since it appears not to know the proper knock sequence
- Replay attacks are not easily thwarted
- Inability to send a reasonable amount of data (asymmetric encryption cannot be used for example)
- Network footprint is noisy – a knock sequence looks just like a port scan to any IDS

Single Packet Authorization Projects

- Two main classes: cipher-based vs. hash-based
- Four SPA projects today:
  - Tumbler
  - Netfilter SPA extension
  - NMRC SPA
  - fwknop

SPA - Network View



SPA 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
- Not well suited to client-side protection
- libpcap vulnerabilities

fwknop-1.8 Release – New Features

- fwknop SPA client integration with Windows (under Cygwin)
- fwknopd SPA server integration with ipfw firewalls on *BSD systems
- gpg-agent integration so passwords can be acquired from the agent (both the fwknop client and server)
- New 'accept defaults' installation mode (makes it easy to package fwknop for the Source Mage Linux distribution)

Windows Integration

- The fwknop installer detects Cygwin environments
- The fwknop SPA client is installed, and all perl modules for the fwknopd server are skipped (a Windows firewall is not yet supported)
- Both symmetric (Rijndael) and asymmetric (GnuPG Elgammal, etc.) algorithms are supported

ipfw Integration

- fwknop rules added to the beginning of the ipfw policy (with a configurable entry point rule number)
- knoptm daemon parses the ipfw policy and removes fwknop rules after the timeout defined by each FW_ACCESS_TIMEOUT variable
- The fwknop client works on *BSD systems
- FreeBSD style init script

ipfw Integration (cont'd)

- Receive SPA packet via libpcap, decrypt, perform replay check, verify access authorization, insert ipfw access rule for protected service

ipfw add 1 pass tcp from <SPA IP> to any 22 keep-state

iptables Integration

- fwknop rules in a custom iptables chain: FWKNOP_INPUT
- strict separation is maintained from any existing policy

```
# iptables -I FWKNOP_INPUT -p tcp --dport 22 -s <SPA
IP> -j ACCEPT
```

fwknop.conf

```
EMAIL_ADDRESSES              mbr@cipherdyne.org;
AUTH_MODE                    PCAP;
PCAP_INTF                    eth1;
ENABLE_PCAP_PROMISC          Y;
PCAP_FILTER                  udp port 62201;
PCAP_PKT_FILE                /var/log/ulogd.pcap;
ENABLE_MD5_PERSISTENCE       Y;
```

fwknop access.conf

SOURCE: ANY;

DATA_COLLECT_MODE: PCAP;

OPEN_PORTS: tcp/22;

#ENABLE_CMD_EXEC: Y;

KEY: <encryptkey>;

GPG_DECRYPT_ID: ABCD1234;

GPG_DECRYPT_PW: <password>;

GPG_REMOTE_ID: 1234ABCD;

#GPG_AGENT_INFO: /tmp/gpg-n7jEPC/S.gpg-agent:18333:1;

FW_ACCESS_TIMEOUT: 10;

fwknop Packet Format

Random data: 445893609198753 2

Username: mbr

Timestamp: 1123247144

Version: 1.8

Action: 1 (access mode)

Access: 123.123.123.123,tcp/22

MD5 sum: wrfuSWoS+py7ppsESNR78A

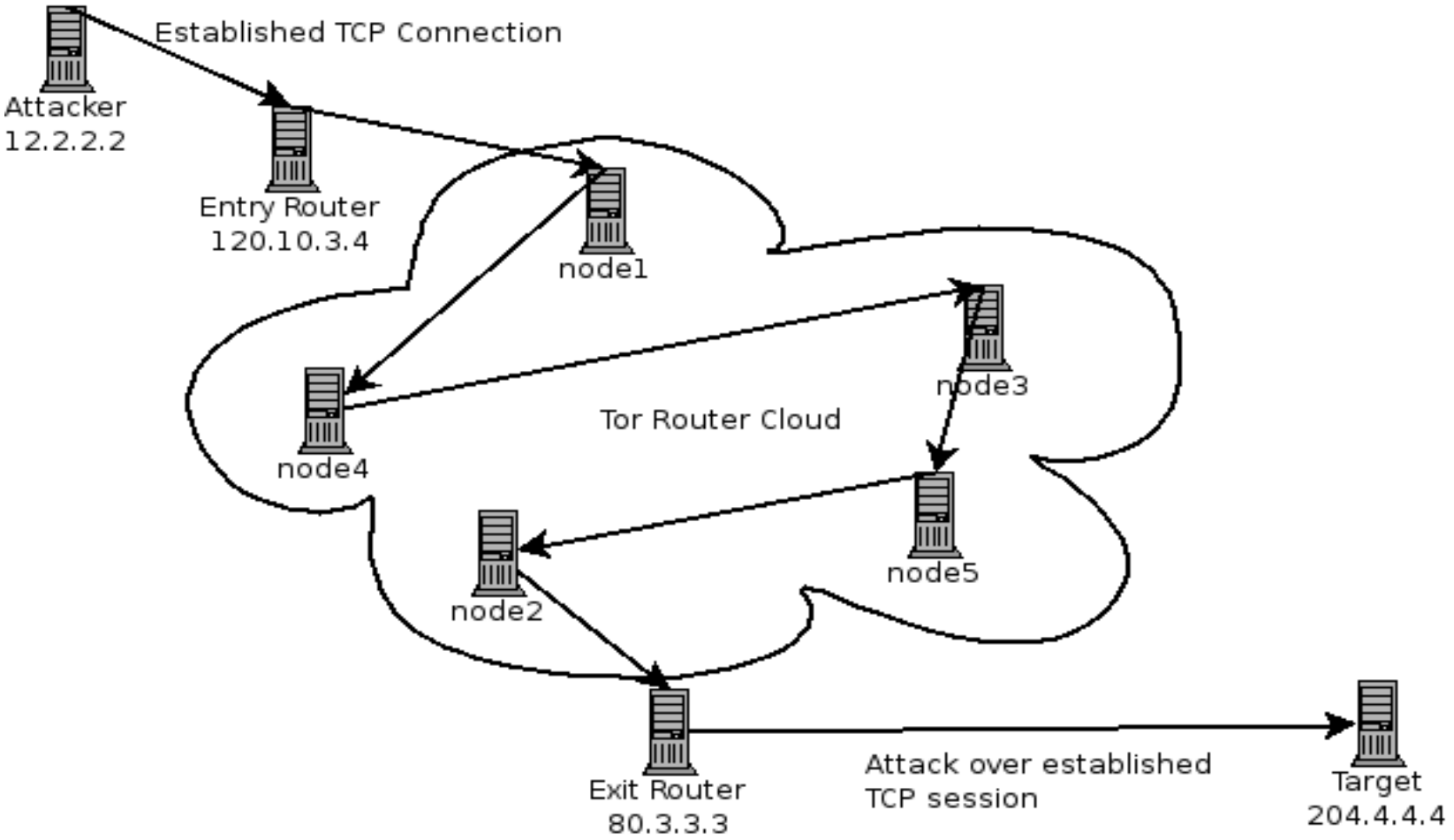
<optional server authentication criteria>

445893609198753 2:mbr:1123247144:1.8:0.0.0.0,tcp/22:wrfuSWoS+py7p psESNR78A

Reducing IDS/IPS Event Load

- Recall Stick/Snot attacks against Snort before stream preprocessor
- Imagine an updated Stick/Snot tool that sends faked attacks over the Tor anonymizing network – works against the stream preprocessor (depending on the signature)
- SPA makes such an attack infeasible because NO session can be established until a valid SPA packet is generated – reduces false positives

Attacks Over Tor

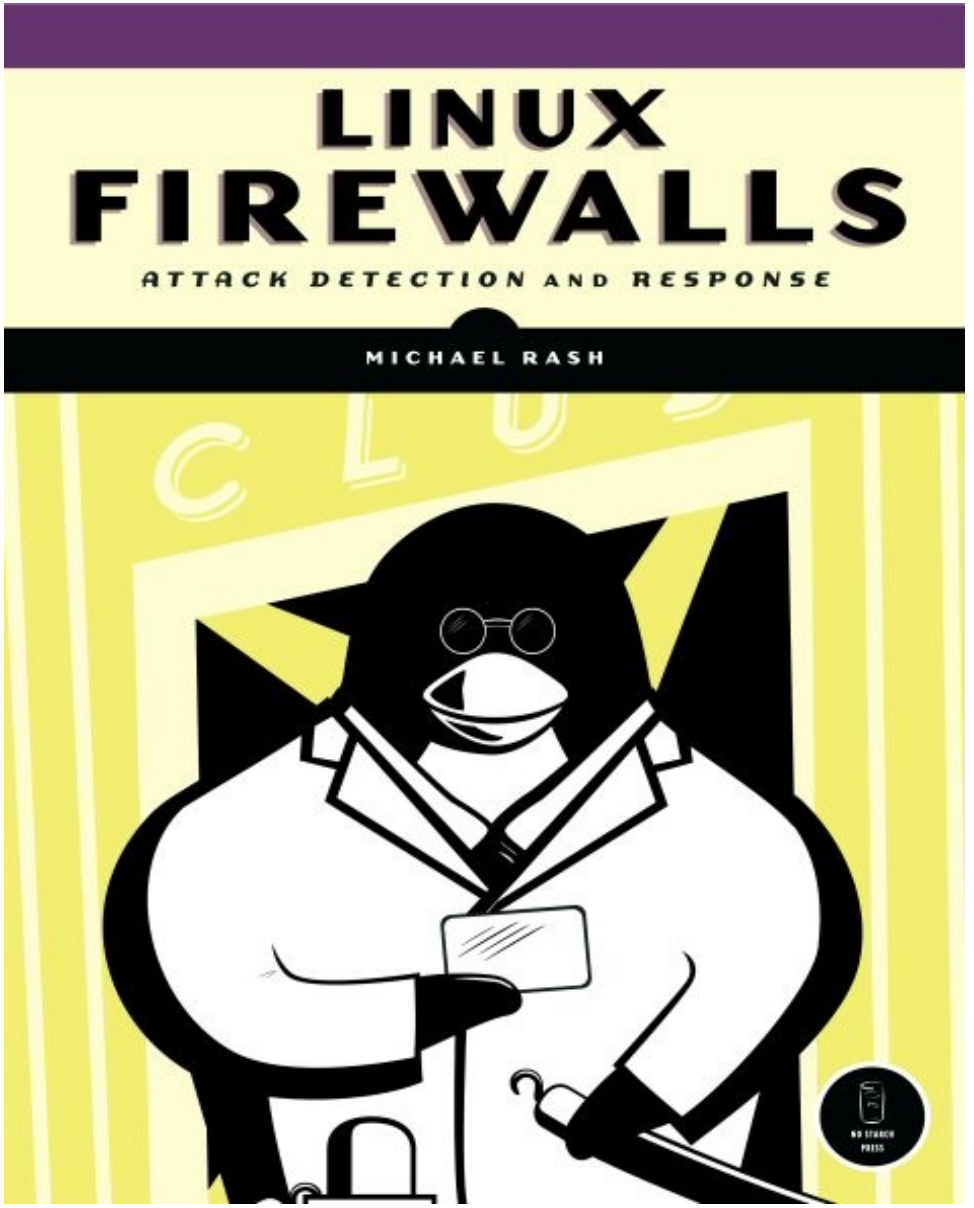


Live Demo...

Passive Authorization Articles

- “Protecting SSH Servers with Single Packet Authorization”, *Linux Journal, May 2007*
- “Single Packet Authorization”, *Linux Journal, April 2007*
- “Single Packet Authorization with fwknop”, *USENIX :login; Magazine, February 2006*
- “Combining Port Knocking and Passive OS Fingerprinting with fwknop”, *USENIX ;login: Magazine, December 2004*

No Starch Press, Sept 2007



Questions?

http://www.cipherdyne.org/

mbr@cipherdyne.org