Security of IP Routing Protocols

GCIH Practical Assignment Version 2.0
for GIAC Certification in
Advanced Incident Handling & Hacker Exploits

Chris Russell

October 7, 2001

Table of Contents

1 PREFACE 1

1.1 Introduction 1

1.2 The Problem 1

1.3 Errors and Omissions 2

2 THE EXPLOIT 2

2.1 Brief Description 2

2.2 Protocols Used & Potential Variants 3

2.3 Operating Systems Affected 3

2.4 References 3

3 THE ATTACK 4

3.1 Network Configuration 4

3.2 Exploit #1: ARP Flood 5
3.2.1 Description of the Attack 5
3.2.2 Protocol Description 6
3.2.3 Signature of the Attack 6
3.2.4 How to Protect Against It 7
3.2.5 Variant #1 7

3.3 Exploit #2: Spoofed RIP Advertisements 8
3.3.1 Description of the Attack 8
3.3.2 Protocol Description 9
3.3.3 Signature of the Attack 9
3.3.4 How to Protect Against It 9
3.3.5 Variant #1 10
3.3.6 Variant #2 10
3.3.7 Variant #3 10

3.4 Exploit #3: Spoofed HSRP 10
3.4.1 Description of the Attack 10
3.4.2 Signature of the Attack 11
3.4.3 How to Protect Against It 12

3.5 Step 4: Attack! 12

4 THE INCIDENT HANDLING PROCESS 12

4.1 Preparation 12

4.2 Identification 13

4.3 Containment 15

4.4 Eradication 16

4.5 Recovery 17

4.6 Followup & Lessons Learned 17

5 APPENDIX A – BACKGROUND INFORMATION: ROUTING PROTOCOLS 18

5.1 Static Route Tables 18

5.2 Interior Gateway Protocols 18
5.2.1 RIP v1 – Routing Information Protocol (RFC 1058) 18
5.2.2 RIP v2 – Routing Information Protocol (RFCs 2453 & 2082) 19
5.2.3 OSPF v2 – Open Shortest Path First (RFC 2328) 20
5.2.4 IBGP v4 – Internal Border Gateway Protocol (RFC 1771-1772) 21
5.2.5 IGRP – Interior Gateway Routing Protocol (Cisco proprietary) 21
5.2.6 EIGRP – Enhanced Interior Gateway Routing Protocol (Cisco proprietary) 22
5.2.7 IS-IS – Intermediate-System to Intermediate-System (ISO 10589) 22

5.3 Exterior Gateway Protocols 23
5.3.1 EGP – Exterior Gateway Protocol (RFC 904) 23
5.3.2 BGP v4 – (External) Border Gateway Protocol (RFC 1771-1772) 23
5.3.3 S-BGP – Secure BGP (Under Development) 24

5.4 Redundant Router Protocols 24
5.4.1 HSRP – Hot Standby Router Protocol (Cisco Proprietary, RFC 2281) 24
5.4.2 VRRP – Virtual Router Redundancy Protocol (RFC 2338) 24

5.5 Related Protocols 25
5.5.1 ICMP Redirect (RFC 792) 25
5.5.2 IRDP – ICMP Router Discovery Protocol (RFC 1256) 25
5.5.3 ARP – Address Resolution Protocol (RFC 826) 26

6 APPENDIX B – PROTOCOL-LEVEL EXPLOITS 26

6.1 Information Gathering 26

6.2 Path Integrity and Redirection 26

6.3 RIP-Specific Exploits 27

6.4 OSPF-Specific Exploits 27

6.5 Using RIP to Bypass OSPF (or other) security 27

6.6 ARP Exploits 28

6.7 BGP Exploits 28

6.8 Authenticated IGPs on Hosts 29

7 APPENDIX C – BEST PRACTICES 29

7.1 Keep It Simple & Secure (KISS) 29

7.2 Document 30

7.3 Policy 31

7.4 Audit 31

7.5 Computer Incident Response Team (CIRT) 31

7.6 Other Recommendations 32

8 BIBLIOGRAPHY 32

1 Preface

1.1 Introduction

This paper focuses on the inherent security of routing protocols and how weak/insecure protocols can be exploited to assist in launching other attacks within a network. It also highlights the dangers of misconfigured network gear. It does not, however, focus on security holes created by bugs in software, but rather focuses on the underlying ramifications of the network protocols as defined by their RFCs.

It used the example scenario in the DEFCON 9 paper as an example exploit, discusses the steps that would be taken to perform the attack, and how to identify and protect against it. In a case study style, it describes an incident handling session to discover and respond to the attack.

Finally, the appendices document key highlights for each of the major IP routing protocols used today, security ramifications, several known exploits against the fundamental protocols, and best practices for securing the network, network devices, and routing protocols.

1.2 The Problem

Information security primary focuses in two areas:

- Secure the network by installing a firewall.
- Secure the computers by hardening the system (configuring security features, disabling unnecessary or insecure network services, etc.) and installing security patches.

However, many facilities fail to consider the security ramifications of other networked devices, such as intelligent routers, switches, and network appliances. Many of these devices contain their own operating systems and complex applications. Indeed, as routers become more complex, properly configuring, administering, and securing them becomes increasingly difficult.

I've seen large IT divisions of Fortune 500 companies that had whole departments dedicated to data security, that deployed state-of-the-art network intrusion detection tools and techniques, and yet failed to properly configure their firewall rules, forgot to password protect their routers, or used insecure protocols that were prone to session hijacking or spoofing.

If these companies had such a hard time getting it perfect, what chance does anyone else have?

By manipulating routing protocols, a hacker can sniff the network, intercept traffic, launch spoof and man-in-the-middle attacks, and open up doors to other security vulnerabilities that might have been otherwise inaccessible.

I was surprised that discussion of potential security exploits of routing protocols was conspicuously missing from the Hacker Exploits class at SANS 2001 in Baltimore. When asked, I was told they weren't included because they just haven't seen many attacks using these techniques in the wild. However, I suspect that times are changing. I've personally observed a

BGP hijacking at a former company (unfortunately, we didn't have a security team at the time and were unable to track it down before the problem was corrected by our ISP). And later this year at DEFCON 9, FX taught a introductory class called "Routing & Tunneling Protocol Attacks" that described how to use insecure protocols to assist with attacks, and similar discussions have taken place at prior DEFCONs.

Tools are being developed to assist in exploiting routing protocols, such as IRPAS and Nemesis.

1.3 Errors and Omissions

While I've done my best to avoid mistakes in this document, I recognize there may be a few technical errors and omissions. Therefore, an updated version of this document will be maintained at http://www.infosecalliance.com/whitepapers/security-of-ip-routing for public use.

2 The Exploit

2.1 Brief Description

As a case study, I am building upon the attack scenario [28] presented by FX at DEFCON 9. An attacker is connected to an internal (private) office LAN and attempting to hack into a database server to access confidential information. However, there are a couple obstacles in his way:

- Attacker is directly connected to a switch and therefore unable to sniff the network.
- The database server is protected behind an internal firewall and all ports to it are blocked.

Attacker performs three network protocol exploits, enabling him to sniff the network, intercept a user's session, and ultimately stumble upon a way of bypassing the firewall's protection.

- Exploit #1: Attacker sends an *ARP Flood* to the switch, slagging its ARP cache and essentially turning it into a hub. He can sniff the network. This is a similar technique (but applied for different results) to CVE candidate CAN-1999-0667, "The ARP protocol allows any host to spoof ARP replies and poison the ARP cache to conduct IP address spoofing or a denial of service." Furthermore, the ability to sniff a network is identified by CVE candidate CAN-1990-0530, "A system is operating in "promiscuous" mode which allows it to perform packet sniffing".

- Exploit #2: Attacker broadcasts *Spoofed RIP Advertisements* to alter route tables and intercept user sessions through the firewall. This vulnerability is identified as CVE-1999-0111: "RIP v1 is susceptible to spoofing".

- Exploit #3: Attacker sends *Spoofed HSRP* messages to switchover the primary firewall to a misconfigured standby firewall, thus giving him complete access into the secure network. There are currently no CVE vulnerabilities files against the HSRP protocol (although there should).

After exploit #3 is complete, Attacker has free access into the secure network, unencumbered by the firewall, and can proceed to hack into the servers using traditional hacking techniques.

2.2 Protocols Used & Potential Variants

Exploit #1 uses spoofed ARP replies. Variants of this attack use spoofed IP packets instead.

Exploit #2 uses spoofed RIP packets. Variants of this attack could use ARP or IRDP instead. If other unauthenticated IGP or EGP protocols were used on the network, then they could be spoofed to achieve similar results. (See §5 below for a description of these protocols.)

Exploit #3 uses spoofed HSRP. If other unauthenticated redundant router protocols were used (such as certain implementations of VRRP), then it could be spoofed for similar results.

2.3 Operating Systems Affected

The three exploits are protocol-based attacks; they are simply using supported features of the protocols as defined by their respective RFCs. They do not exploit coding bugs in the operating system, applications, or firmware, and therefore can not be "fixed" by simply applying security patches.

Minimizing the threat of these vulnerabilities requires careful network design, careful selection and configuration of critical network protocols, and targeted network intrusion detection.

2.4 References

The following tools and references describe these specific exploits:

Exploit #1:

- *Some TCP/IP Vulnerabilities*, http://www.staff.washington.edu/dittrich/talks/agora/index.html, Dave Dittrich
- Dsnif tools arpspoof and macof (and accompanying man pages), http://www.monkey.org/~dugsong/dsniff, Dug Song.
- ARP0c tool (using the undocumented command line switch "-f"), http://www.phenoelit.de/arpoc, Phenoelit.
- Angst, an active sniffer (with integrated ARP flooding), http://angst.sourceforge.net, Patroklos Argyroudis.

Exploit #2:

- *Protecting Network Infrastructure at the Protocol Level*, http://www.netw3.com/documents/Protecting_Network_Infrastructure.htm, Curt Wilson.
- Nemesis Packet Injection tool suite, http://jeff.wwti.com/nemesis, Mark Grimes.
- Internet Routing Protocol Attack Suite (IRPAS) tools & documenation, http://www.phenoelit.de/irpas/docu.html, Phenoelit.

Exploit #3

- *Routing & Tunneling Protocol Attacks*, http://www.phenoelit.de/stuff/routing.pdf, Presentation at DEFCON 9, FX / Phenoelit.

- Internet Routing Protocol Attack Suite (IRPAS) tools & documenation, http://www.phenoelit.de/irpas/docu.html, Phenoelit.

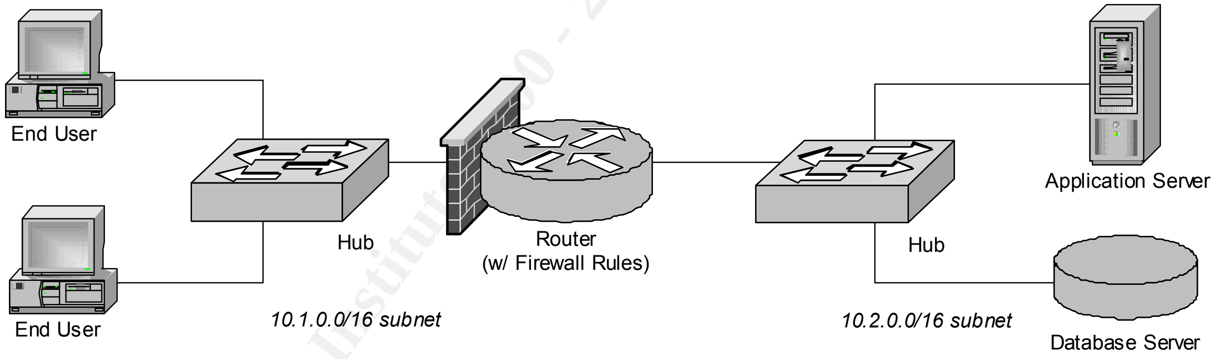
3 The Attack

3.1 Network Configuration

Originally, the network was divided into two subnets:

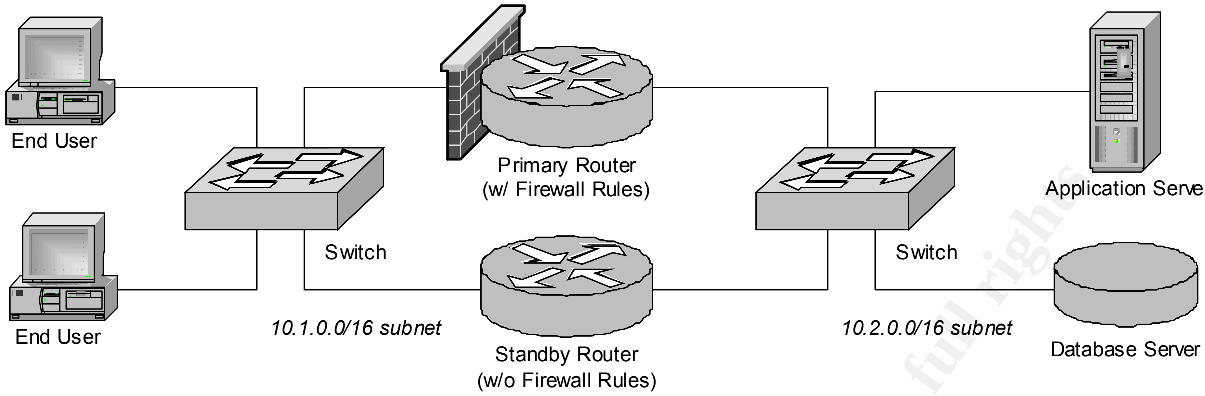
- The office subnet is 10.1.0.0/16, used for desktop PCs and local printers.
- The data center network is 10.2.0.0/16, used for servers and databases.

A router connects the two subnets, and tight firewall rules setup on the router protect the data center systems. Only individual ports were opened to data center systems on an as-needed basis. For example, port 80 was opened to the application server so users could access it, but the database server was completely blocked off since only the data center servers are allowed to access it directly.



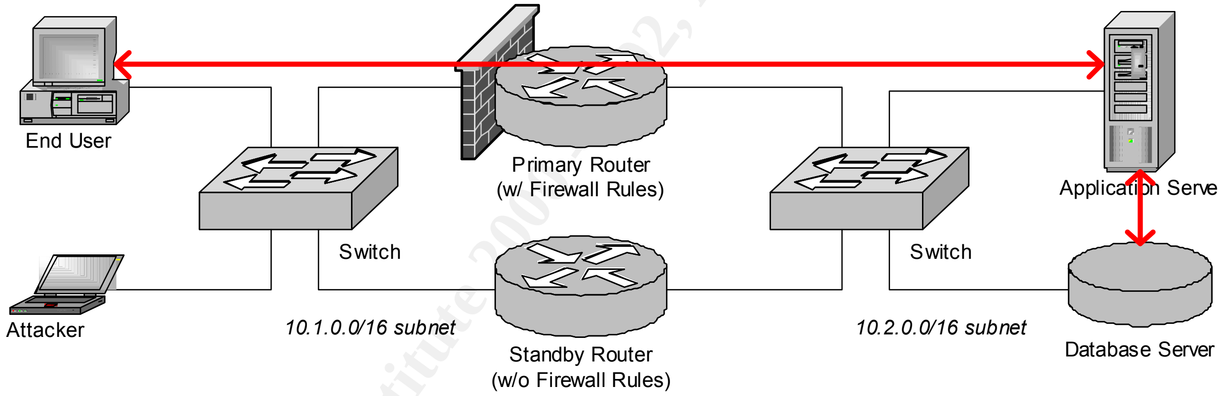
Since the data center network was protected behind a firewall, the IT staff felt it safe to use the "r-commands" (rdist, rcp, rsh, etc.) and NFS, reasoning that the firewall implicitly blocked ports 513 (rlogin), 514 (rexec), and 2049 (NFS).

Over time, the hubs were upgraded to switches to improve aggregate performance on the network, and a second router was added for fault tolerance. Using Cisco's HSRP redundant router protocol, one router was designated the primary router and the other the standby router. Under normal circumstances, the primary router was active (and used to route traffic between the two subnets) and the standby router was passive (routing disabled). However, if the primary router ever failed for any reason, then the standby router would automatically become the primary and take over all routing.



HSRP was tested extensively before installing the standby router, to ensure it works properly. Unfortunately, the network administrator was rushed and therefore forgot to setup firewall rules on the standby router before installing it!

Unfortunately, the misconfiguration of the standby router went undetected. Normally, only the primary router is active; therefore, security is enforced and the vulnerability is undetected. Under *normal* circumstances, the standby router isn't used unless the primary router fails.



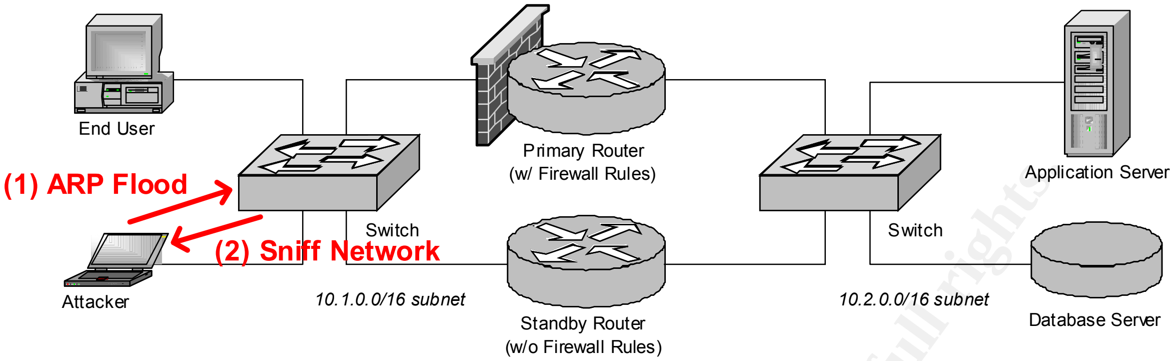
Then one day, Attacker decides to hack into the database to access some confidential information.

3.2 *Exploit #1: ARP Flood*

3.2.1 Description of the Attack

Attacker's first problem is sniffing the network. The network switch automatically filters out all unicast packets that are not specifically addressed to Attacker's MAC address.

Therefore, he sends a flood of ARP replies ("I am") with random IP and MAC addresses, hoping to overload the switch. The switch accepts the unsolicited replies and inserts them to its ARP cache. The cache quickly fills up with bogus MAC addresses, flushing out older addresses in FIFO (first in first out) or LRU (least recently used) order.



Since all of the valid MAC addresses for devices on the network have been flushed out of the cache, the switch no longer knows how to route the packets and therefore degrades down to behaving like a hub, forwarding copies of all packets it receives into all of its ports.

Attacker can now sniff the network, searching for passwords, trust relationships, etc., so long as he continues flooding the switch with spoofed ARP replies.

Attacker uses the following command to generate the continual (nonstop) ARP flood:

```
# ARP0c2 -I eth0 -f
```

3.2.2 Protocol Description

The ARP protocol is defined in the network layer, encapsulated directly inside the Ethernet frame. It uses Ethernet type 0x0806, specified in bytes 12:13 in the Ethernet or 802.3 header.

For TCP/IP, the ARP header is 28 bytes long:

Bytes 0:1	Hardware type (1 = Ethernet)
Bytes 2:3	Protocol type (0x0800 = TCP/IP)
Byte 4	Hardware address length (6 = 48 bit Ehternet MAC addresses)
Byte 5	Protocol address length (4 = 32 bit IP addresses)
Bytes 6:7	Opcode (2 = Reply)
Bytes 8:13	Sender Ethernet address
Bytes 14:17	Sender IP address
Bytes 18:23	Target Ethernet address
Bytes 24:27	Target IP address

For ARP flooding, the sender and target Ethernet and IP addresses are all filled with random addresses.

Further descriptions of the ARP protocol and potential exploits are detailed in §5.5.3 and §6.6 below.

3.2.3 Signature of the Attack

A tcpdump listing of ARP0c2 produces:

```
0:0:0:0:0:0 > a3:3c:dd:de:16:51 null I (s=4,r=0,R) len=24
              0604 0002 a33c ddde 1651 772c 8667 5661
              6e4f 27c9 772c 8667
0:0:0:0:0:0 > b9:c4:29:20:be:14 null I (s=4,r=0,R) len=24
              0604 0002 b9c4 2920 be14 8726 aef9 21e6
              675a 560a 8726 aef9
0:0:0:0:0:0 > 95:34:e8:ab:85:5f null I (s=4,r=0,R) len=24
              0604 0002 9534 e8ab 855f d60b c52d 6b34
              7b92 fd34 d60b c52d
0:0:0:0:0:0 > 57:27:54:16:3b:da null I (s=4,r=0,R) len=24
              0604 0002 5727 5416 3bda 3be8 d35b cf3b
              b426 454a 3be8 d35b
0:0:0:0:0:0 > 59:2d:f4:dd:8b:ca null I (s=4,r=0,R) len=24
              0604 0002 592d f4dd 8bca e851 f654 8571
              e682 a53d e851 f654
```

This is a very unusual signature for ARP messages. Close inspection of the Ethernet frame's data payload (listed in hex) reveals it to be a properly formed ARP reply, but from the output of tcpdump, this is obviously a crafted datagram. (To be honest, I've never seen tcpdump output like this before!)

Even though ARP spoofs generated by ARP0c2 can be easily identified, it would be simple to modify it to generate normal looking ARP datagrams (albeit containing random IP and MAC addresses).

Other indicators include unusually heavy network traffic on the network and bandwidth degradation, since the switch is now operating more like a hub.

3.2.4 How to Protect Against It

Some switches allow static ARP tables to be defined. The MAC addresses of all known devices would be preloaded into the switch; however, this is exceptionally impractical.

It is possible that a layer 3 switch may be resistant to this specific attack, since it switches based on network addresses (IP addresses) rather than link addresses (Ethernet MAC addresses). However, it may be similarly vulnerable to layer 3 attacks, such as IP flooding? An intelligent switch with subnet information and anti-spoof detection could theoretically detect this and protect against it.

Finally, installing a network intrusion detection system (NIDS) probe can detect this attack, either by statistical analysis or more simply by detecting ARP replies containing erroneous network addresses (in this case, networks other than 10.1.0.0/16).

3.2.5 Variant #1

Instead of generating ARP replies, sending a flood of IP datagrams (either TCP or UDP) with random IP and MAC addresses accomplishes the same effect as an ARP flood.

This may be accomplished using the Dsnif command:

```
    # macof -r -i eth0 -n 100000
```

which produces the following tcpdump listing:

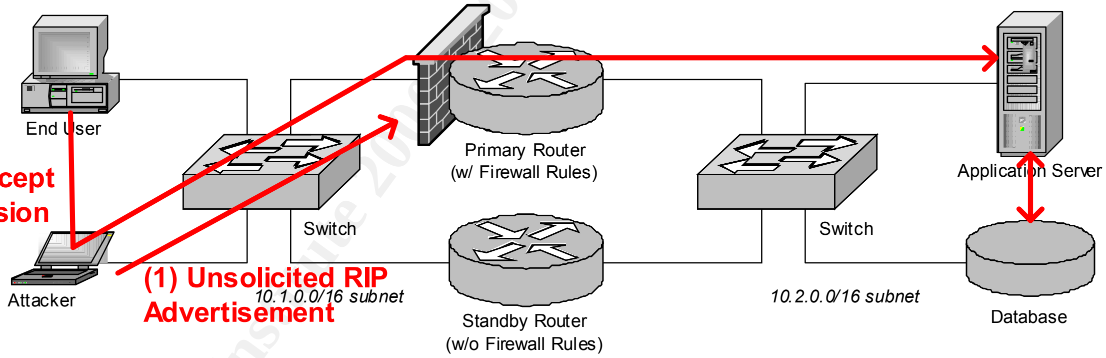
```
174.183.209.156.26355 > 133.36.213.136.50073: . 0:0(0) win 65535 (DF) [tos 0x10]  (ttl 64, id 0)
165.113.175.249.25334 > 67.134.251.70.20810: . 0:0(0) win 65535 (DF) [tos 0x10]  (ttl 64, id 0)
155.161.109.154.1731 > 136.39.6.168.2007: . 0:0(0) win 65535 (DF) [tos 0x10]  (ttl 64, id 0)
144.181.52.123.54255 > 103.147.17.67.11487: . 0:0(0) win 65535 (DF) [tos 0x10]  (ttl 64, id 0)
4.111.234.237.57042 > 35.13.219.53.34409: . 0:0(0) win 65535 (DF) [tos 0x10]  (ttl 64, id 0)
```

3.3 Exploit #2: Spoofed RIP Advertisements

3.3.1 Description of the Attack

After sniffing the network, Attacker decides to intercept all traffic from the end user (at 10.1.7.11) to the application server (at 10.2.0.5). He has a few methods available to choose from, but selects RIP since the hosts are configured for RIP v1 and it is unlikely to setoff any alarms.

Attacker sends the end user a RIP packet specifying a new route directly to address 10.2.0.5 with a metric (hop count) of 1. The end user's computer gobbles up the unsolicited route advertisement and proceeds to route all future packets to the application server through Attacker as a gateway. Attacker may then perform man-in-the-middle attacks by reading and potentially altering the packets before forwarding them on to their final destination.


End User
(2) Intercept the Session
Attacker
Switch
(1) Unsolicited RIP Advertisement
10.1.0.0/16 subnet
Primary Router (w/ Firewall Rules)
Standby Router (w/o Firewall Rules)
Switch
10.2.0.0/16 subnet
Application Server
Database

In this example, Attacker is only intercepting the traffic in one direction. In similar fashion, he may also setup routes with the primary router to intercept and route traffic back in the other direction.

The following command would issue a RIP v1 command to route all traffic from host 10.1.7.11 to the server 10.2.0.5 through the Attacker's computer at 10.1.123.123:

```
    #./nemesis-rip -c 2 -V 1 -a 1 -i 10.2.0.5 -m 1 -V 1 \
        -S 10.1.123.123 -D 10.1.7.11
```

Or, to generate a RIP v2 route with proper netmask information, use:

```
    #./nemesis-rip -c 2 -V 2 -a 1 -i 10.2.0.5 -k 0xfffffff-m 1 -V 1 \
        -S 10.1.123.123 -D 10.1.7.11
```

Attacker must also configure his computer to route the packets onward. Otherwise, this would result in an inadvertent routing black hole and the packets never reach their destination.

3.3.2 Protocol Description

RIP is a very simple protocol, described in more detail in §5.2.1 below. It uses UDP port 520, and the header is:

Bytes 0:1	Command (2 = Response, or Advertise Routes)
Bytes 2:3	Not used (set to 0x0000)
Bytes 4+	RIP v1 entry table

Each entry in the entry table is specified as follows:

Bytes 0:2	Address family
Bytes 3:4	Route tag (set to 0x0000 for RIP v1)
Bytes 5:8	Target IP address
Bytes 9:12	Target subnet mask (set to 0x00000000 for RIP v1)
Bytes 13:16	Next hop IP address (set to 0x00000000 for RIP v1)
Bytes 17:20	Metric (hop count)

The Next hop address specifies the router's IP address. If the next hop value is 0 (which is always the case for RIP v1), then the router's address defaults to the sender's IP address of the RIP datagram.

3.3.3 Signature of the Attack

A tcpdump of the RIP v1 command produces:

```
10.1.123.123 > 10.1.7.11.route: rip-resp 1: [family 1: 0000 0500 020a 0000 0000 0000 0000 0000
0001](1) (DF) [tos 0x18]  (ttl 254, id 122)
```

A tcpdump of the RIP v2 command produces:

```
10.1.123.123 > 10.1.7.11.route: rip-resp 1: [family 1: 0000 0500 020a ffff ffff 0000 0000 0000
0001](1) (DF) [tos 0x18]  (ttl 254, id 123)
```

These are all normally formatted RIP packets. Perhaps the best way to detect these as illegal (spoofed) RIP packets is to filter for any RIP responses (but not requests) originating from unexpected IP addresses, such as any address that is not from a known router on the network segment.

However, this will not catch RIP v2 packets that spoof the source IP address from a known router but uses the Next hop value to specify the Attacker as the router. Filtering these packets requires a significantly more detailed analysis into the application layer of the datagram, which is beyond the capabilities of tcpdump.

3.3.4 How to Protect Against It

To protect against this exploit:

- Disable route discovery protocols on host machines. Use static route tables instead.
- Only use cryptographically secure IGP and EGP protocols on routers, such as OSPF or RIP v2 with MD5 authentication.
- Disable IRDP, if possible, to prevent similar exploits using IRDP route insertion.

3.3.5 Variant #1

Instead of exploiting layer 3 route protocols, the network traffic can be intercepted using layer 2 ARP cache poisoning, similar to the technique used by hunt for TCP session hijacking across a switch.

However, this technique is prone to detection, since modern operating systems tend to generate spoofed packet alerts whenever they detect ARP replies advertising their own IP address.

3.3.6 Variant #2

Instead of using an IGP protocol like RIP, it is possible to manipulate the route tables using IRDP route insertion. This technique enables Attacker to selectively intercept data between two hosts (or between a host and a router) without having to intercept (and forward) all of the traffic from the router.

This approach is superior to ARP spoofing since it is generally more stealthy and less likely to setoff and alarms. However, not all hosts are configured to accept IRDP.

CVE identifies a special case of this exploit (when using DHCP) as CVE-1999-0875, "DHCP clients with ICMP Router Discovery Protocol (IRDP) enabled allow remote attackers to modify their default routes".

3.3.7 Variant #3

If the hacker is connected link network between neighboring exterior routers, then this technique can be applied to EGP protocols (such as BGP) to intercept traffic between autonomous systems, thus enabling man-in-the-middle attacks across the Internet!

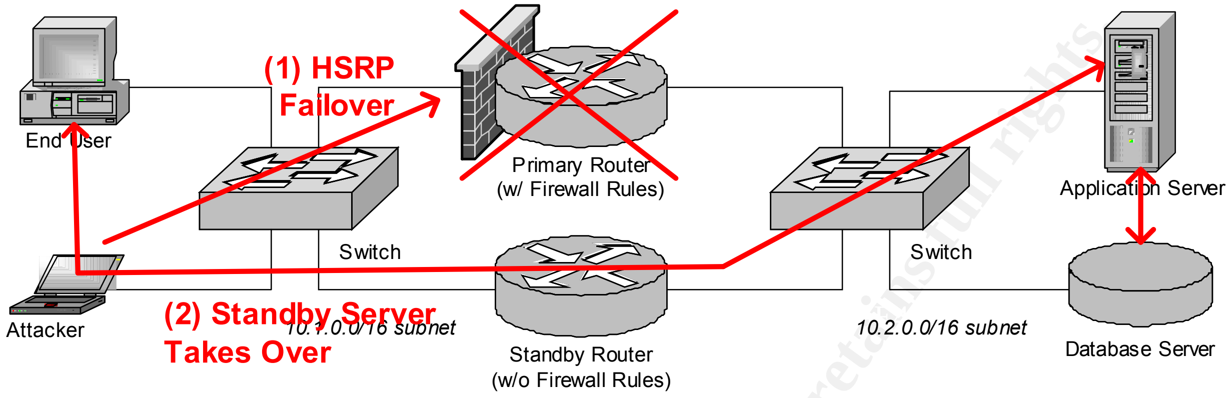
This potential vulnerability is described in more detail in §6.7 below.

3.4 Exploit #3: Spoofed HSRP

3.4.1 Description of the Attack

Attacker thoroughly scans the primary router but can find no vulnerabilities or misconfigurations. However, in the process, he detects a hot standby router using HSRP. Scanning it, Attacker discovers that there are no firewall rules enabled on the standby server! At last the break he was looking for!

Attacker sends a spoofed HSRP message to the primary router (at 10.1.0.1) to essentially take it offline. Once offline, the standby router becomes the new primary router and automatically starts to route all traffic between the two networks.



There is no longer any firewall protection for the data center network, and Attacker is free to use traditional tools and techniques to hack into the systems.

If the primary router's address is 10.1.0.1, then the following command would be used to put it into standby mode, and hence failover to the other router:

```
     # while (true)
     do
          ./hsrp -d 10.1.0.1 -v 10.1.0.2 -a cisco -g 1 -i eth0
          sleep 3
     done
```

3.4.2 Signature of the Attack

A tcpdump listing of hsrp produces:

```
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 80)
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 80)
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 82)
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 82)
```

Or, using 'tcpdump –x' to include the packet data (in hex) produces:

```
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 80)
                        4500 0030 005c 0000 8011 2598 0a01 0001
                        0a01 0001 07c1 07c1 001c 0000 0001 1003
                        ffff 0100 6369 7363 6f00 0000 0a01 0002
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 80)
                        4500 0030 005c 0000 8011 2598 0a01 0001
                        0a01 0001 07c1 07c1 001c 0000 0001 1003
                        ffff 0100 6369 7363 6f00 0000 0a01 0002
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 82)
                        4500 0030 005e 0000 8011 2596 0a01 0001
                        0a01 0001 07c1 07c1 001c 0000 0000 1003
                        ffff 0100 6369 7363 6f00 0000 0a01 0002
10.1.0.1.1985 > 10.1.0.1.1985: udp 20 (ttl 128, id 82)
                        4500 0030 005e 0000 8011 2596 0a01 0001
```

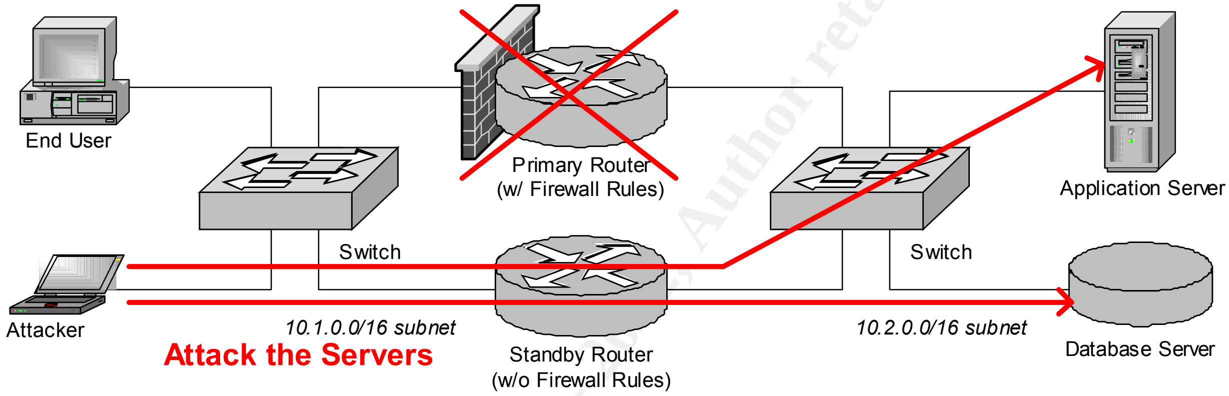
```
0a01 0001 07c1 07c1 001c 0000 0000 1003
ffff 0100 6369 7363 6f00 0000 0a01 0002
```

3.4.3 How to Protect Against It

Again, only use authenticated protocols. HSRP is inherently insecure. Therefore, if possible, use VRRP with MD5 authentication instead.

3.5 Step 4: Attack!

Attacker now has full access into the data center network and is busy hacking away. He breaks r-command trust relationships, cracks Unix password files, and even finds a few backoffice servers with security bugs since they haven't been patched for years!



4 The Incident Handling Process

"An ounce of prevention is worth a pound of cure."

4.1 Preparation

The IS department was security savvy and already implemented a number of the best practices recommendations listed in §7 below.

- They disabled all unnecessary protocols and network services from their routers and servers.
- They thoroughly documented their network and its required capabilities.
- They created a cross-functional Change Management team to supervise, approve, and schedule all server and network changes.
- All employees were requires to sign acceptable use policies, and warning banners are displayed from all networked devices (including servers, routers, and individual workstations).
- They even created a "volunteer" CIRT team composed of system engineers, network engineers, management, and human resources.
- Pre-established policies (in writing) authorizing the CIRT team to take specific actions when necessary, such as shutting down servers and sniffing the network.

One of the CIRT members was predestinated as the lead, and everyone in the IS department was trained to notify him in case of a suspected security breach. If he is unavailable, then his supervisor or the VP of Technology is to be notified in his absence.

Two CIRT "jump kits" were created in advance, each of which contained:

- A voice recorder and several notebooks to record a step-by-step journal of activities.
- CDs containing bootable Linux, Solaris, and Windows operating systems, along with copies of diagnostic, forensic, and backup tools: ls, ps, df, lsof, netstat, dd, tar, etc.
- One large IDE drive and one large SCSI drive, with drive cables and "Y" power splitters.
- A small laptop (an old Pentium 233MHz system, only costs $250 from eBay) with network sniffing and NIDS software pre-installed.
- A small network hub (*not* a switch) and Ethernet cables (both straight-through and cross-over cables).
- A number of small plastic bags and a permanent marker, for storing evidence.
- A blank case file to start filling out when an incident occurs.

A network intrusion detection system (NIDS) probe is installed outside the corporate network, to detect intrusion from the Internet. However, no probes are currently installed onto the office or data center networks to detect unauthorized activity from the inside.

The help desk is responsible for maintaining a contact list of all vendors, ISPs, and support contracts.

4.2 Identification

Exploits #1 and #2 were performed during the day and went completely undetected. A few users noticed some network congestion and slowdown during the ARP flood, and a couple computers crashed, but that's nothing new. The IS team was prompt to reboot the crashed systems so the users can get back to work.

Later that evening at 9pm, Attacker performed exploit #3 and switched over to the standby router, thus gaining access to the data center network. As a side effect, the failover generated an SNMP trigger and paged a network engineer (who happened to be at home at the time, what a slacker!).

The engineer logged in remotely from home to confirm that the standby router switched over properly and that the network was still up. After testing the connection with a few pings and sprays, he was pleased with the results and glad that he didn't have to drive back to work that evening. Thankful that he installed the standby router, he could deal with the problem in the morning.

Several days went by without incident, and then at 8pm the primary router failed again. This time, the network engineer drove back to work, frustrated and determined to figure out why the router is malfunctioning. He arrived by 9pm, re-checked the error logs, and ran basic diagnostics on the failed router. Convinced that everything should be in working order, he attached a network probe to both networks (using a couple small hubs to splice them in), started recording using tcpdump, and (at 9:45pm as recorded in the logs) switched the failed router back as the primary router.

Checking the error logs, he was shocked to find a number of security alerts generated by the firewall, reporting several hosts on the office network directly connected to the database server and other critical systems.

Concerned, he contacted the CIRT lead immediately (on his cel phone) and filled him in on the situation. The lead asked the engineer to temporarily stop troubleshooting and write down everything that he's done and observed so far, to the best of his memory.

Around 10:30pm, the CIRT lead arrived at work and met with the engineer. Together, they made a quick assessment of the situation:

- The last security alert reported by the firewall occurred at 9:48pm, just shortly after the routers were switched back.
- The primary router is still functioning propertly.
- Other servers in the data center appear to be up and operational.
- Therefore, there appears to be no immediate threat requiring immediate action.

The CIRT lead opened a new case file, recorded these observations into the journal, and entered the following pieces of evidence (with each item uniquely numbered for reference):

- Printouts of the log files from the routers.
- The notebook used by the network engineer to record his observations and actions.

Realizing from the firewall logs that several servers appear to have been compromised, the CIRT lead notified the rest of the CIRT members and assembled a small team. Everyone was assembled and briefed by 11:30pm and they setout to locate the source of the intrusion. Each member of the CIRT team recorded all of their actions in a written log, and they signed and dated the bottom of each page with a witness, if available.

The system administrators were sent to investigate server and application logs, while the network engineers attempted to track down the source of the source of the packets.

All of the offending packets originated from 10.1.123.123.

- This IP address was issued to Attacker's computer by the DHCP server. Confirming the MAC address of the packets against the MAC address recorded in the DHCP server logs helped further confirm that the address was probably not spoofed. These logs were printed out, numbered as evidence, and handed over the CIRT lead to log and store into the case file.
- By querying the TrackIt! database (used for fixed asset tracking), the CIRT team was able to determine who's computer had that particular MAC address.

By 1am, the CIRT team determined that the firewall rules were not enabled on the standby router. However, they could not determine if the router was always configured that way or if the hacker changed the configuration himself.

4.3 Containment

The CIRT team decided to shut down the standby server, thereby temporarily containing the security breech (not withstanding any potential damage or backdoors installed onto the servers, if any). However, before powering off the router, they output the full router configurations into text files. Both the save configurations (in NVRAM) and current working configurations (in RAM) were output, just in case they differed. The files were then signed (but not encrypted) by the CIRT lead's private PGP key, printed out, numbered, logged, and entered as evidence into the case file. Finally, the passwords were then changed on both routers and the standby server was powered down.

It is now 2am in the morning, and the CIO is now called at home and notified of the situation.

The CIRT team encounters their first major glitch. They need to make a bitwise backup of Attacker's hard drive before powering up his computer. However, Attacker is using a laptop with a 2.5" IDE drive, which uses a different type of cable than normal 3.5" drives. The jump kit doesn't include a 2.5" <> 3.5" IDE adapter cable. They'll have to wait until morning. Therefore, they store the laptop into a large plastic bag, tag it as evidence, log it, and lock it up securely.

Logs are gathered from each server, digitally signed (again, using the CIRT lead's PGP key), and logged as evidence. The attacker's login account is temporarily disabled. And finally, comfortable that everything is reasonably under control, the CIRT team breaks to go home and catch a couple hours' sleep before resuming in the morning.

At 8am the following morning, the team reassembles with the CIO for a quick briefing. The CIO then notifies the appropriate members of senior staff: human resources, the CEO, and COO.

By 10am, the CIRT team is able to locate a 2.5" <> 3.5" IDE cable adapter and so removes Attacker's hard drive from the laptop and installs it as a secondary drive on a desktop system. First booting into Linux (to be as unobtrusive as possible), they investigate the partition table. There are three partitions:

- NTFS (the active / boot partition)
- Linux
- Linux swap

For thoroughness, all three partitions were backed up onto a spare hard drive using dd. Then, after being backed up, the original hard disk was bagged, tagged, and logged into evidence.

Linux was then booted from a CD in the CIRT jump kit and the backed up partition was mounted. The partition contained a number of hacking tools, including nessus, IRPAS, crack, nmap, whisker, a (*non*-kernel level) rootkit for Linux, and other tools. Unauthorized use of these tools on the corporate network is specifically prohibited (by anyone other than the CIRT team).

The Legal and Human Resources departments were notified, and they in turn contacted the FBI's High Tech Crimes Task Force.

4.4 Eradication

Now, the focus of the CIRT team is to assess potential damages, perform a risk assessment, and determine how best to eliminate any backdoors, trojans, or vulnerabilities.

All system passwords are securely updated, and user passwords are flagged to be changed the next time each user logs in.

Analyzing the tools on Attacker's Linux partition, it doesn't appear to have any Windows or NT tools or keystroke loggers. However, it does contain a root kit.

Therefore, scripts are written to execute on all Unix systems in order to:

- List out the current working state of each system, including running processes (ps –ef), disk space (df –k), network connections (netstat, netstat –r, etc.), open files (lsof), device inventory (hinv), logins (who), etc.
- Search for rootkit files, searching by filename, checksum, and/or modification date.
- Audit all executables with the suid flag enabled.

Just to be safe on Windows machines, the virus definition files are updated and all hard drives are scanned.

Using information gathered from the firewall alerts, log files, and scripts, a list of affected systems was compiled, and each of these systems was completely backed up onto tape (again, using dd) and entered into as evidence. (If possible, it would help if HR could try to get a written statement from Attacker detailing which systems he affected.)

All servers were then evaluated using port scanning (nmap) and vulnerability scanning (nessus, whisker, etc.) software.

Before modifying any servers currently in production, or restoring any systems back into production, the CIRT team needs to coordinate with the Change Management team. Together, they plan an approach to restore the systems and rotate them back into production:

- First, surgically replace any rootkit affected files and return the systems quickly back into production.
- Then, for each affected system, reinstall the operating system and applications onto a separate machine, copy over the data files from the server (rather than from backup tape), and rotate it into production (replacing the old server).

- Create a project (scheduled through Change Management) to eliminate the use of r-commands, NFS, telnet, and other insecure protocols, and keep systems updated with current security patches.

The first step is done immediately and the servers are quickly returned to production.

4.5 Recovery

At this point, the facility is up and running. However, the affected systems still need to be reinstalled from scratch, just to make sure that nothing is missed.

Rather than installing Linux from scratch, a master disk is created, patched, tested, certified, and then cloned for each server using the command:

```
# find / -depth –mount | cpio -pdumv /mnt/newdisk
```

This helps accelerate the process of restoring systems with the added bonus of standardizing installations and performing consistent levels of testing. Next, the any special applications are loaded, configuration files copied (and checked), and data files copied. Finally, a domain expert and/or "power user" manually tests and verifies that the system is working correctly before recommending to Change Management to have it rotated into production.

The systems are then monitored by the IT group and selected power users to ensure there are no glitches. Ideally, the old hard drives from the systems are persevered for a period of time just in case any problems arise.

4.6 Followup & Lessons Learned

Finally, the case file is closed and a summary report is written up by the CIRT lead. Custody of the file is then handed off from the CIRT lead to the CIO.

A port mortem meeting of the incident gave rise to the following suggestions and observations:

- A network intrusion detection system (NIDS) probe should also be installed *within* the corporate network.
- Route discovery protocols should be disabled on all desktop PCs.
- Authenticated protocols should be used whenever possible: VRRP instead of HSRP, OSPF instead of RIP, etc.
- When installing new gear (such as the standby router), slow down and spend the time to thoroughly test and know what you're doing. It's tough, but everyone must focus on developing realistic expectations and schedules. Otherwise, cutting corners and making mistakes is inevitable.
- It is not obvious how aggressively to install security updates. While they are necessary to eliminate known vulnerabilities, they may also cause instability and detrimental side effects. This requires further discussion and policy.
- Tripwire all the servers!!! That way you'll know for sure what was affected by a hack and what wasn't!

- It would be nice to purchase more voice recorders, spare hard drives, cables (especially 2.5" IDE cables!), and enCase for write blocking and forensically backing up hard disks.

5 Appendix A – Background Information: Routing Protocols

The Internet is too large to be managed as a single network. Instead, it is broken up into a number of independent, connected administrative domains called autonomous systems (AS). Each AS is administered by a single entity and consists of a collection of routers that coordinate routing information using dynamic gateway protocols.

- Routing *within* an autonomous system is accomplished using Interior Gateway Protocols (IGP), such as RIP, OSPF, IBGP, IGRP, EIGRP, and IS-IS.
- Routing *between* autonomous systems is accomplished using Exterior Gateway Protocols (EGP), such as EGP and BGP.

Other routing protocols are used to ensure high availability, redundant link paths or devices, etc.

5.1 Static Route Tables

The simplest and perhaps safest method of specifying routes is to set them manually in static route tables. This way, dynamic route discovery protocols are not needed and may be disabled.

Even if dynamic route protocols are needed for the routers, hosts can typically make do with just a single default route. In these cases, keep it simple and only use static routes for the hosts. Otherwise, it may be simple to manipulate the route tables on the hosts.

5.2 Interior Gateway Protocols

Interior Gateway Protocols (IGPs) are used to dynamically manage and optimize routing within a single AS. It is possible (albeit inadvisable) to run multiple IGPs simultaneously within a single AS. This is often done as a temporary measure to ease a migration from an older protocol (such as RIP) to a newer one (such as OSPF).

5.2.1 RIP v1 – Routing Information Protocol (RFC 1058)

RIP v1 is one of the oldest and most commonly used IGP routing protocol. It is supported on virtually every platform that uses TCP/IP and is trivially easy to setup. (It is so easy to setup that I've encountered several companies that were using it without even realizing it!)

A RIP process runs on each router or host. There are two types of RIP processes:

- An *active* process (1) advertises its routes and (2) listens for advertisements from other routers.
- A *silent* (passive) process only listens for advertisements but never advertises its own routes.

RIP is a *distance-vector routing protocol*. In other words, every route is assigned a metric value (also called the "hop count") that specifies the number of routers a packet must traverse before reaching the target network. An active router advertises its routes via UDP broadcast. Other RIP processes listen to these advertisements, compare the advertised routes against their current route tables, and select the routes with the lowest metric values. New routes are added, and better routes replace those with larger hop counts.

RIP processes may also solicit routers by unicasting or broadcasting a UDP request for active processes to advertise their routes immediately. Hosts may perform this during startup in order to quickly initialize its route table.

The longest number of hops supported by RIP is 15. Routes with a metric value larger than 15 are discarded. Furthermore, routes typically expire after 180 seconds. Therefore, active routers advertise their routes periodically, such as every 30 seconds, to prevent them from being automatically deleted from route tables.

Unfortunately, RIP is the easiest routing protocol to hack.

- It is UDP-based (typically using port 520) and stateless.
- It does not use internal sequence numbers.
- It allows unsolicited route advertisements. In other words, spoofed advertisements (responses) are accepted and processed even though no requested it.
- RIP datagrams are not authenticated.

Therefore, it is trivially easy to spoof RIP. There are no sequence numbers to predict, sessions to hijack, authentication passwords to sniff, trust relationships to spoof, or crypto keys to crack. By spoofing RIP, it is possible to manipulate route tables on routers and hosts. This vulnerability is referenced by CVE-1999-0111, which states very simply: *"RIP v1 is susceptible to spoofing"*.

It is also easy to identify (active) RIP-enabled routers and download their route tables by sending them a request message. A single router may be requested via UDP unicast, or all routers on a network may be requested via a single UDP broadcast datagram.

5.2.2 RIP v2 – Routing Information Protocol (RFCs 2453 & 2082)

RIP v2 improves upon the functionality of RIP v1:

- v2 adds a netmask to each route, thus eliminating ambiguities that existed with v1.
- v2 can advertise routes for other routers (using the "Next Hop" field).
- v2 supports *multicasting* periodic advertisements to IP address 224.0.0.9. v1 required *broadcasting* periodic advertisements, which caused RIP packets to be sent to all machines, even if they weren't running RIP listeners.
- v2 adds cleartext authentication!
- RFC 2082 (an extension to RIP v2) adds MD5 authentication!

Unfortunately, there are a few limitations to authentication:

- RFC 2453 (which defines the RIP v2 protocol) only specifies **cleartext authentication**. This method is weak and easy to break. Essentially, shared passwords are transmitted in the clear between RIP processes, making them vulnerable to being intercepted over the network. Once intercepted, packets can be trivially spoofed with little more effort than RIP v1.
- RFC 2082 defines a superior method of authentication for RIP v2 using **MD5** hashes and shared keys that are never transmitted over the network. While this is an excellent method of authentication, it is defined as an extension to RIP v2 and not included in RFC 2453. Thus, it may be more difficult to gain universal support for this implementation.
- RIP v1 processes ignore authentication altogether.
- Authentication is optional.

Whenever possible, RIP should be used with MD5 authentication enabled. This approach transmits a one-way cryptographic hash with each RIP message. The hash is generated by appending an authentication key (up to 16 characters in length) to the message prior to hashing with the MD5 algorithm. Thus, the authentication key is never transmitted. Only the resulting hash is added to the RIP message.

When a listening RIP process receives a message, it regenerates the hash using the same steps and compares hash values. If they match, it the RIP message was sent by an authorized server. Otherwise, the RIP message was either spoofed or altered, and thus is discarded. As an added bonus, MD5 authenticated messages include a sequential number to protect against replay attacks.

Unfortunately, most facilities don't enable any authentication for RIP, rendering it no more secure than RIP v1. In fact, unauthenticated RIP v2 can be *more* dangerous than v1, since it allows spoofing multiple levels of routers, not just the ones on the local segment. Forged route advertisements can be propagated across multiple networks within the autonomous system, thus making it even easier for a hacker to gain access and redirect traffic across network segments.

Unlike v1, there are no CVE vulnerabilities defined for RIP v2, even though it can be even more dangerous!

5.2.3 OSPF v2 – Open Shortest Path First (RFC 2328)

As opposed to RIP, which is a distance-vector routing protocol, OSPF is a *link-state routing protocol*. Each OSPF process floods the network with link state advertisements (LSAs) that identifies all of the interfaces attached to the router, along with various metrics and other relevant information. Each router accumulates LSA information to build a map of the entire autonomous system, and then uses the shortest path first (SPF) algorithm to calculate the shortest route between networks.

False LSAs are contested (fight back?) by the other routers, making it more difficult (although not impossible) to spoof routes. Therefore, it is typically easier to intercept and modify LSA transmissions between routers, so the target router will accept the modified LSA.

OSPF is a complex protocol to setup. Therefore, it may be prone to errors in its configuration. If the hacker understands OSPF better than the sysadmin that setup the network, then he may have the upper hand.

OSPF does not use TCP or UDP but rather defines its own protocol directly on top of IP, using protocol number 89.

OSPF defines three forms of authentication:

- **Null Authentication.** LSAs are not authenticated and therefore can be easily spoofed. Only a simple checksum is generated to detect unintentional data corruption, but this can be easily defeated or spoofed.
- **Simple Password.** A shared password (8 bytes long) is embedded in the clear in LSA messages. This method is trivial to defeat by sniffing the network and intercepting passwords or altering LSAs in transit.
- **Cryptographic Authentication.** Similar to MD5 authentication in RIP v2, a shared authentication key is hashed with the LSA message to produce an cryptographically secure MD5 hash. This hash is then transmitted with the LSA and validated by all receiving OSPF processes using the same authentication key. The key itself is never transmitted over the network. Furthermore, a monotonically increasing sequence number protects against reply attacks, although there does exist a brief window of opportunity for replay attacks until the sequence actually increments.

OSPF only provides authentication, not confidentiality. Therefore, hackers can sniff the network for LSA messages in order to map out the network topology. This is extremely useful information for reconnaissance.

Unlike RIP v1, there are no CVE vulnerabilities defined for OSPF, even though it can also be spoofed if used in null or simple password authentication mode. Furthermore, §6 below defines a number of OSPF-specific vulnerabilities and attacks.

5.2.4 IBGP v4 – Internal Border Gateway Protocol (RFC 1771-1772)

BGP is typically used as an exterior gateway protocol (EGP) to exchange route information between external neighbors of different autonomous systems (AS). However, it may also be used as an interior gateway protocol (IGP) for internal routing between internal neighbors within the same AS. When used in this way, it is called Internal BGP, or IBGP.

For the most part, IBGP shares many of the same vulnerabilities as BGP. In practice, however, it is probably easier to hack IBGP since the routers more likely to be connected via shared network segments rather than dedicated lines. Therefore, it is easier for a hacker to gain link network access between two IBGP routers and forge UPDATE messages.

Refer to §5.3.2 and §6.7 below for a more detailed description of BGP and potential security exploits.

5.2.5 IGRP – Interior Gateway Routing Protocol (Cisco proprietary)

IGRP is another distance-vector routing protocol developed by Cisco in the mid 1980's to overcome a number of limitations in RIP v1, including:

- RIP's small hop count limit (15).
- RIP's single metric is too limiting.
- IGRP supports multipath routing (for increased aggregate bandwidth), weighted load balancing (where faster lines in a multipath link get a larger percentage of the aggregate traffic), and automatic failover between lines if one goes down.

Unfortunately, it is a proprietary protocol and does not support authenication. Therefore, IGRP packets can be easily spoofed and route tables manipulated on participating routers and hosts.

5.2.6 EIGRP – Enhanced Interior Gateway Routing Protocol (Cisco proprietary)

Enhanced IGRP implements a number of performance enhancements over IGRP and the Diffusing Update Algorithm (DUAL) to combine several capabilities of link state and distance vector routing protocols.

EIGRP also adds two (optional) forms of authentication:

1. **Plain Text Authentication.** A shared is embedded in the clear in EIGRP messages. This method is trivial to defeat by sniffing the network and intercepting passwords or altering messages in transit.
2. **MD5 Authentication.** Similar to MD5 authentication in RIP v2 and OSPF, a shared authentication key is hashed with the LSA message to produce an cryptographically secure MD5 hash. This hash is then transmitted with the message and validated by all receiving EIGRP processes using the same authentication key. The key itself is never transmitted over the network.

If MD5 authentication is not used, then EIGRP packets can be trivially spoofed or altered in while in transit.

Automatic redistribution enables IGRP routes to be automatically imported into EIGRP, and visa versa. However, this can circumvent EIGRP authentication, since IGRP does not implement authentication.

Also, EIGRP does not support confidentiality, so it is possible for a hacker to sniff the network and build a network map.

5.2.7 IS-IS – Intermediate-System to Intermediate-System (ISO 10589)

IS-IS is another link state routing algorithm designed for routing ISO/CLNP (Connectionless Network Protocol) packets. Like OSPF, it uses a shortest path first algorithm to determine the best routes from accumulated link state information.

However, it does not appear to support authentication, so presumably IS-IS packets can be spoofed. Fortunately, this is not a very popular protocol.

5.3 Exterior Gateway Protocols

Exterior Gateway Protocols (EGPs) are used to dynamically manage and optimize routing between AS's. EGPs are the connective tissues that connect together and define the Internet. If hacked, it could bring the Internet to its knees.

5.3.1 EGP – Exterior Gateway Protocol (RFC 904)

EGP is one of the first exterior gateway protocols, used to exchange route information between autonomous systems.

Two routers first establish adjacency between each other by exchanging Hello and I-H-U (I Head You) messages. Once adjacency is determined, then the two routers are considered "neighbors" and can exchange route information.

EGP is on top of IP, protocol #8. It is stateful and therefore includes a sequence number. However, it is does not support authentication, so packets can be spoofed.

EGP has been mostly replaced by BGP v4 for routing over the Internet.

5.3.2 BGP v4 – (External) Border Gateway Protocol (RFC 1771-1772)

BGP v4 is the primary EGP protocol used to exchange route information between autonomous systems across the Internet. It includes mechanisms for preventing route loops between autonomous systems.

BGP transmits four kinds of messages during a session: open, update, notification, & keepalive. Once routes are exchanged, then only changes to the route information (via update messages) are sent.

BGP uses TCP port 179, does not include its own sequence numbers (besides those used by TCP), and does not appear to support secure authentication.

Ironically, RFC 1771 specifies authentication for BGP v4 but does not specify what specific authentication methods are supported (cleartext, MD5, etc.) and how they would be applied. RC 2385 defines a method for providing MD5 authentication to the underlying TCP transport in BGP sessions, as an interim method of providing secure authentication.

Therefore, Cisco has added support for MD5 authentication of peers and generates random sequential numbers to increase the difficultly of hijacking sessions.

On a practical note, most communications between autonomous systems on the Internet are performed over direct lines. Therefore, it is physically difficult to get link access between the two BGP servers, rendering spoofing impractical.

According to Nick Feamster [31],

> "The forging of the opening of a BGP session is extremely difficult, because responses from the spoofed interface must also be disabled. Additionally, the end host will also be expecting full BGP tables, and route flapping, the frequent addition and withdrawal of a route from the routing tables, will surely occur if the session opening is not performed properly.
>
> However, it is possible to insert forged BGP "UPDATE" messages (i.e., route updates) into an existing BGP session between two peers since the only sequence number included in the UPDATE packets are the TCP sequence numbers. This means that a malicious AS could potentially spoof the BGP UPDATE messages; spoofing BGP update messages boils down to essentially performing spoofing of TCP messages. (Note that this attack is not possible against modern versions of Cisco's IOS.)"

This attack would be made against a pre-established EBGP-MULTIHOP session between BGP routers.

5.3.3 S-BGP – Secure BGP (Under Development)

In order to address the security weaknesses inherent in BGP, there are efforts to develop a Secure Boarder Gateway Protocol (S-BGP) that utilizes PKI. However, there is no standard yet, and acceptable of S-BGP may be difficult to obtain.

5.4 Redundant Router Protocols

HSRP and VRRP are redundant router protocols that enable multiple routers over a multi-access link to share the same virtual IP address. One router is the elected to be the active (or master) router while the others are standby routers. If the master router fails, then one of the standby routers becomes the new master, thus ensuring high availability.

5.4.1 HSRP – Hot Standby Router Protocol (Cisco Proprietary, RFC 2281)

HSRP is a proprietary redundant router protocol developed by Cisco. The protocol only supports cleartext authentication with an 8 byte password. This password can be easily sniffed by machines on the network, and therefore is not a secure protocol.

HSRP packets can be easily spoofed to force a switchover to a standby router, including switchover to a rogue router. This can result in traffic redirection, man-in-the-middle attacks, packet black holes, and DoS.

5.4.2 VRRP – Virtual Router Redundancy Protocol (RFC 2338)

According to RFC 2338: "The Virtual Router Redundancy Protocol (VRRP) is designed to eliminate the single point of failure inherent in the static default routed environment. VRRP specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VRRP routers on a LAN. The VRRP router controlling the IP address(es) associated with a virtual router is called the Master, and forwards packets sent to these IP addresses. The election process provides dynamic fail-over in the forwarding responsibility should the Master become unavailable. Any of the virtual router's IP addresses on a LAN can then be used as the default first hop router by end-hosts. The advantage gained from using VRRP is a higher availability default path without requiring configuration of dynamic routing or router discovery protocols on every end-host."

Thus, VRRP is a similar protocol to Cisco's HSRP.

VRRP supports three forms of authentication:

- **No Authentication:** VRRP protocol is not authenticated and is highly vulnerable to being spoofed.
- **Simple Text Password:** A shared password (8 bytes long) is embedded in the clear in LSA messages. This method is trivial to defeat by sniffing the network and intercepting passwords or altering VRRP messages in transit.
- **IP Authentication Header:** Similar to MD5 authentication in RIP v2, a shared authentication key is hashed with the VRRP message to produce an cryptographically secure MD5 hash. This hash is then transmitted with the message and validated by all receiving VRRP processes using the same authentication key. The key itself is never transmitted over the network.

VRRP protects against attacks across networks by always setting the TTL value to 255 and validating upon receipt. Therefore, must attacks against VRRP must be launched from the local network segment and will not survive across a router.

5.5 Related Protocols

5.5.1 ICMP Redirect (RFC 792)

An ICMP Redirect is used to advertise a better route under the following situation, as described in RFC 792: "The gateway sends a redirect message to a host in the following situation. A gateway, G1, receives an internet datagram from a host on a network to which the gateway is attached. The gateway, G1, checks its routing table and obtains the address of the next gateway, G2, on the route to the datagram's internet destination network, X. If G2 and the host identified by the internet source address of the datagram are on the same network, a redirect message is sent to the host. The redirect message advises the host to send its traffic for network X directly to gateway G2 as this is a shorter path to the destination. The gateway forwards the original datagram's data to its internet destination."

ICMP Redirects can be easy spoofed, although they typically require 8 bytes from the original (redirected) datagram.

5.5.2 IRDP – ICMP Router Discovery Protocol (RFC 1256)

IRDP is an extension to ICMP for discovering neighboring routers on a network segment.

Routers periodically announce themselves by multicasting IRDP router advertisements, and hosts discover the routers by listening for them.

A host may also multicast a solicitation request for all routers to immediately announce themselves. Each router responds to the solicitation by multicasting an advertisement. This is typically done with a host starts up.

Solicitations are multicast to the all-routers multicast group (224.0.0.2) or broadcast to 255.255.255.255. Advertisements are multicast to the all-systems multicast group (224.0.0.1) or broadcast to 255.255.255.255.

IRDP currently does not support any form of authentication. Therefore, any system with link access to the network may spoof IRDP packets and masquerade as the default router, as indicated by CVE-1999-0875, "DHCP clients with ICMP Router Discovery Protocol (IRDP) enabled allow remote attackers to modify their default routes." IRPD may also be used add or override other routes by advertising the route with a higher preference level.

5.5.3 ARP – Address Resolution Protocol (RFC 826)

ARP is technically not a routing protocol. However, forged ARP replies can be used to redirect traffic on layer 2 switches or essentially turn switches into routers. These exploits are described further in §6.6 below.

ARP is an unauthenticated protocol defined in the network layer, protocol number 1. It is trivial to spoof.

6 Appendix B – Protocol-Level Exploits

This section illustrates are a number of exploits against the fundamental routing protocols. These examples do not rely upon bugs or platform-specific vulnerably but rather exploit the underlying weaknesses in the protocols as defined by their RFCs.

6.1 Information Gathering

IGP routing protocols may be sniffed over the network to help map out the network, including all routing devices and the overall network topology. Several protocols (such as RIP) support soliciting for routing information via UDP broadcast or multicast.

This is important reconnaissance information, and all IGP protocols allow this.

6.2 Path Integrity and Redirection

Unauthenticated (or weakly authentication) routing protocols, ICMP redirects, IRDP advertisements, and ARP cache poisoning may be exploited to alter the path in which datagrams are transmitted through a network, giving a hacker the ability to:

- Intercept and sniff confidential data.
- Spoof data and exploit trust relationships.
- Perform a man-in-the-middle attacks and alter data in transit.

- In some cases, create DoS black holes and routing loops that consume data without routing it onward.

ICMP redirects are particularly interesting. Since ICMP packets are routable, they can be sent long distance from a across the Internet! However, they do typically require the first 8 bytes of the packet to be redirected. Can this information be forged?

6.3 RIP-Specific Exploits

New routes can be added by advertising that route with a spoofed RIP message. Existing routes can be deleted by advertising the route with a metric of 16, or updated by advertising a new route with a better metric (such as 1) than the existing route.

A black hole (DoS) can be created by advertising a metric 1 route through a non-existing router. All traffic will be redirected to the non-existent router (and thus lost), and routers will even issue ICMP redirects to re-route existing traffic to the black hole!

6.4 OSPF-Specific Exploits

Curt Wilson [12] describes a few known attacks specifically targeted against OSPF's "fight back" mechanism:

> "**Max Age Attack:** The maximum age of a LSA is one hour (3600). Attacker sends LSA packets with maxage set. The original router that sent this LSA then contests the sudden change in age by generating a refresh message in a process called "fight-back". Attacker continually interjects packets with the maxage value for a given routing entity which causes network confusion and may contribute to a DOS condition.
>
> **Sequence++ Attack:** Attacker continually injects a larger LSA sequence number, which indicates to the network that it has a fresher route. The original router contests this in the "fight back" process by sending it's own LSA with an even newer sequence number than the attackers sequence number. This creates an unstable network and could contribute to a DOS condition.
>
> **Max Sequence Attack:** The maximum sequence number 0x7FFFFFFF is injected by an attacker. Attackers router then appears to be the freshest route. This creates the same "fight-back" condition from the original router – IN THEORY. In practice, they found that in some cases, the MaxSeq LSA is not purged and remains in the link state database for one hour, giving an attacker control for that time period."

6.5 Using RIP to Bypass OSPF (or other) security

Routers and routing daemons can be configured to redistribute routing information between IGPs (such as RIP and OSPF) or between an IGP and EGP (such as RIP and BGP). However, this would enable an attacker to use an insecure protocol (such as RIP) to advertise false routes across OSPF or BGP, bypassing their normal security!

6.6 ARP Exploits

ARP packets can be used to redirect traffic on layer 2 switches by broadcasting ARP replies, declaring that a specific MAC & IP address is connected to a given port on the switch.

Unsolicited ARP replies are accepted by the switch and added to the ARP cache, thus redirecting the flow of traffic for that specific host.

The effect is only temporary, since the ARP cache will be corrected as soon as the spoofed host transmits across the switch. Therefore, it is necessary to flood the switch with a periodic stream of forged ARP replies.

This attack is identified by the candidate CVE (under consideration) CAN-1999-0667, "The ARP protocol allows any host to spoof ARP replies and poison the ARP cache to conduct IP address spoofing or a denial of service."

Another exploit is to flood a switch with ARP replies containing random mac & IP addresses, thus overflowing the switch's internal ARP cache. This may either crash the switch or essentially turn it into a hub,

6.7 BGP Exploits

Routing on the Internet relies upon the BGP protocol. Any security vulnerabilities in the protocol could potentially cripple the Internet.

Bradley & Aceves [30] notes a number a potental BGP vulnerabilities, including:

> "There are a number of vulnerabilities that allow a strategically placed intruder to fabricate, modify, replay, or delete routing information. With these capabilities an intruder can compromise the network in a number of ways. The modification or fabrication of routing updates allows an intruder to reconfigure the logical routing structure of an internet, potentially resulting in the denial of network service, the disclosure of network traffic, and the inaccurate accounting of network resource usage. Specific attacks include:
>
> - An intruder can initiate a BGP connection with an authorized BGP speaker. The result of successfully establishing a BGP peer connection by an intruder is the full participation in the routing computation.
> - Any intruder, using the IP spoofing attack, can fabricate a half-duplex BGP session masquerading as an authorized BGP speaker. Given the half-duplex nature of this attack it is not clear that it poses a serious threat.
> - An intruder located on a subnet through which BGP links pass can arbitrarily delet BGP traffic. Due to the reliable transport protocol connection, the effect of this attack will be a denial of service due to the connection being interrupted or disconnected.
> - An intruder located on a subnet through which BGP links pass can, using a TCP session hijacking attack, arbitrarily fabricate, modify, delete, or replay routing information while masquerading as an authorized BGP speaker.

- Any authorized BGP speaker can arbitrarily fabricate, modify, delete, or replay routing information while masquerading as another authorized BGP speaker.

> The vulnerabilities these attacks exploit is the lack of access control, authentication, and integrity of BGP message contents."

Authentication is only performed during the OPEN sequence, and authentication is not widely accepted or implemented. Once a session is open, then forged update messages can be injected. BGP does not implement sequence numbers, although the TCP sequence numbers have to be properly forged.

Fortunately, as stated in §5.3.2 above, most communications between autonomous systems on the Internet are performed over direct lines. Therefore, it is physically difficult to get link access between the two BGP servers, rendering spoofing impractical.

While it is possible to filter out route advertisements from unrecognized IP addresses or AS's, this is typically not done by many major backbone providers (such as Sprint, AT&T, UUNet, etc.) since it would inconvenience the smaller ISPs that utilize their backbones. [31]

6.8 Authenticated IGPs on Hosts

So you're using authenticated IGPs for all your routers and hosts, and think you're secure? The question is, how well protected is the authentication key on the hosts? Can an employee determine the authentication key on his computer or laptop? Can a hacker get it if s/he breaks into the employee's computer?

If the authentication key is obtained, then the protocol can be broken and spoofed. It's a *shared* secret… all parties typically use the same key. Therefore, it is critical to adequately secure all devices that share the secret.

7 Appendix C – Best Practices

The following recommendations are recommended.

7.1 Keep It Simple & Secure (KISS)

Avoid using route discovery protocols on non-routing gear, such as workstations, file servers, and network appliances. Instead, use static routes whenever possible. Default routes are A Good Idea™. Use VIPPR and BGP to overlay link redundancy and high availability on top of the default route.

Carefully select what protocols you enable on hosts, routers, servers, and other networked devices. Only enable those protocols that you truly need *and* use. (For example, you might think you need SNMP, but if you're not using it, then you probably don't.) Spend the time to learn how to properly configure and secure every protocol in use. If possible, test everything in an isolated testbed environment with security settings enabled prior to releasing it for final installation.

Disable all protocols and special features that you don't need or use: NetBEUI, broadcast forewarding, source routing, unnecessary ICMP messages, etc. If you don't have Macintoshes, then disable AppleTalk. If you aren't using QoS, then disable it. Keep It Simple.

Favor secure protocols over insecure. Use OSPF w/ MD5 authentication over RIP. If RIP is necessary, then use RIP v2 with MD5 authentication. Use secure SNMP v3 instead of v1, ssh instead of telnet, ssync instead of rdist, etc. Avoid RIP v1 and weak cleartext authentication like the plague – it's too easy to hack.

Avoid transferring route data between the IGP and BGP protocols. This can expose a backdoor around authentication.

Password protect your network gear, including routers and switches. (I know this sounds like an obvious recommendation. However, I've seen major international companies that forgot to password protect any of their routers, leaving them wide open! Sometimes, it's the obvious things that are overlooked.)

Principal of Least Privilege (POPL): Only issue administrative access to network equipment on a need to know basis. Physically secure the routers in the telco closets so prying hands can't get at them. Remember, if just one router is hacked, then the authentication keys may be revealed and enable route advertisements to be spoofed.

7.2 Document

Thoroughly document the design of your network, including:

- Network diagrams identifying critical gear, the network topology, link connections, etc.
- Routing protocols used: IGPs and EGPs.
- Network- and transport-layer protocols supported: TCP/IP, IPX/SPX, AppleTalk, NetBEUI, etc.
- Special features: High availability / redundancy protocols, QoS, CoS, VoIP, etc.
- Try to briefly describe why each of these protocols and features are used and by whom. That way, you can quickly determine the potential impact if you need to modify, upgrade, or disable them in the future.

Store all configuration files for network devices into a versioning file repository, such as CVS, RCS, or ClearCase. This enables quickly tracking and diagnosing changes made and rolling back to previous versions.

Consider periodically dumping and archiving dynamic state information, such as route tables, link-state databases, ARP caches, etc. This information can be used as a baseline for comparison if the devices are hacked or exploited in the future.

Label all network cables and patch panels. This makes it so much easier to physically trace and investigate network anomalies!

Maintain electronic and hard copy phone lists to ISP contacts and VIPs.

Document all passwords, and establish policy on when, how, and who to retrieve them.

7.3 Policy

Implement Change Management (CM) policies requiring all changes to the network to be tested, approved, scheduled, and announced prior to being released. Archive all configuration files into a versioning file repository, such as CVS, RCS, or ClearCase.

Foster an environment that supports reporting security vulnerabilities. Don't use it to bang people over the head, but to learn and strengthen security. Training & education.

Only issue administrative passwords from a single source, and log every time they are issued. Prohibit sharing of passwords or distributing administrative passwords between peers, even they believe that both parties are authorized to have the password.

All equipment must display warning banners restricting logins to authorized access only and defining appropriate use. All employees much sign NDA, appropriate use, etc.

Periodically monitor for new security patches. Establish policies with Change Management to determine how to weigh the risks of applying new security patches (and possibly deteriorating the stability of the system) vs. leaving a system unpatched (and potentially vulnerable to hacking). In any case, thoroughly test all patches before releasing them, preferable tested in an isolated testbed environment.

7.4 Audit

Use network sniffers (such as tcpdump) periodically to audit the networks and assure everything is setup and configured properly. This is an excellent way of detecting misconfigured (or accidentally enabled) services and devices.

Scan the network using port scanning (nmap) and vulnerability detection software (such as nessus and whisker).

Ensure proper backups are being made, including backups of network device configuration files and the CM respository.

7.5 Computer Incident Response Team (CIRT)

If you're lucky enough to have an official CIRT team, then keep them trained and familiar with config. They should participate in configuration management meetings.

If not lucky enough, then at least setup a reporting structure for security incidents and chain of responsibility for incident handling & response. Create a grassroots/voluneeer/reserves cross-functional CIRT team, even if it's not their day job.

Create an emergency communications plan. Emergency action plan. KNOW WHAT AUTHORIZATIONS ARE REQUIRED TO TAKE ACTION (for passive sniffing, active packet injection, reconfiguring systems, taking systems offline, etc.), AND GET THAT POLICY IN WRITING!!! Otherwise, your ass may be grass.

Keep copies of passwords and encryption keys on physical and electronic form offline but accessible given the proper authorization and procedures.

Unless it's a trivially small facility, have two-way handheld radios for sys admins for communicating. These are useful when tracing wiring, network closets, etc.

7.6 Other Recommendations

Batz [32] made the following security recommendations regarding BGP security at Black Hat Briefings 1999:

> "A properly filtered network will not be implicitly affected by a misconfigured or abused network. All sites should filter RFC1918 netblocks. All sites that have implementations that support authentication should use it. Any IGP routes should be static routes that are inserted into BGP directly. i.e IOS's 'network' command. Do NOT redistribute RIP into BGP. Be careful if you absolutely must redistribute BGP routes into your IGP. There are many design alternatives available. BGP communities directly affect the traffic on your network. Use them sparingly and only when specifically necessary. Do NOT assume the obscurity of your communities is adequate protection. Access to your core routers gives an excessive amount of power to an attacker, both over your network and over your peers. Do NOT filter solely based upon AS_PATH information. This is more trust than anyone should have in any network. Wait with bated breath for IPSec to be more widely implemented."

Other recommendations include filtering port 179, using MD5 authentication (if supported), and filtering traffic by prefix list and neighbor.

8 Bibliography

The following reference materials were used in generating this report.

1. *Attacks Using Exploits or Vulnerabilities in Network Protocols*, http://pfft.net/stacy/essay/smurf.html, Stacy Brown.

2. *Beginning BGP*, http://www.mentortech.com/learn/welcher/papers/bgp1.htm, Peter J. Welcher.

3. *BGP Technical Tips*, http://www.cisco.com/warp/public/459/18.html, Cisco online documentation.

4. *Common Vulnerabilities and Exposures Database*, http://www.cve.mitre.org/cve/, The MIRTE Corporation.

5. *Configuring GateD*, http://www.nexthop.com/techinfo/manuals/o_config_guide/config_guide.shtml#UnicastProtocolStmts, NextHop Technologies, Inc.

6. *Improving Security on Cisco Routers*, http://www.cisco.com/warp/public/707/21.html, Cisco online documentation.

7. *Interior Gateway Routing Protocol*, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/igrp.htm, Cisco online documentation.

8. *Internet Routing Protocol Attack Suite (IRPAS) Documenation*, http://www.phenoelit.de/irpas/docu.html, Phenoelit.

9. *ISO Protocols (IS-IS)*, http://www.protocols.com/pbook/iso.htm, Protocols.com

10. *Neighbor Router Authentication*, http://www.cisco.com/univercd/cc/td/doc/product/software/ios113ed/113ed_cr/secur_c/scprt5/scrouter.htm, Cisco online documentation.

11. *OSPF Security Analysis and Intrusion Detection*, http://www.ietf.org/proceedings/99nov/slides/ospf-sec/sld001.htm, Frank Jou, Feiyi Wang, F. Gong, & X. Wu.

12. *Protecting Network Infrastructure at the Protocol Level*, http://www.netw3.com/documents/Protecting_Network_Infrastructure.htm, Curt Wilson.

13. *Protocol Numbers*, http://www.iana.org/assignments/protocol-numbers, IANA.

14. *RFC 1058, Routing Information Protocol*, http://www.ietf.org/rfc/rfc1058.txt, C. Hedrick.

15. *RFC 2453, RIP Version 2*, http://www.ietf.org/rfc/rfc2453.txt, G. Malkin.

16. *RFC 2082, RIP-2 MD5 Authentication*, http://www.ietf.org/rfc/rfc2082.txt, F. Baker & R. Atkinson.

17. *RFC 2328, OSPF Version 2*, http://www.ietf.org/rfc/rfc2328.txt, J. Moy.

18. *RFC 1771, A Border Gateway Protocol 4 (BGP-4)*, http://www.ietf.org/rfc/rfc1771.txt, Y. Rekhter, T.J. Watson, & T. Li.

19. *RFC 1772, Application of the Boarder Gateway Protocol in the Internet*, http://www.ietf.org/rfc/rfc1772.txt, Y. Rekhter, T.J. Watson, & P. Gross.

20. *RFC 904, Exterior Gateway Protocol Formal Specification*, http://www.ietf.org/rfc/rfc0904.txt, D.L. Mills.

21. *RFC 2281, Cisco Hot Standby Router Protocol (HSRP)*, http://www.ietf.org/rfc/rfc2281.txt, T. Li, B. Cole, P. Morton, & D. Li.

22. *RFC 2338, Virtual Router Redundancy Protocol*, http://www.ietf.org/rfc/rfc2338.txt, S. Knight et. al.

23. *RFC 792, Internet Control Message Protocol*, http://www.ietf.org/rfc/rfc0792.txt, J. Postel.

24. *RFC 1256, ICMP Router Discovery Messages*, http://www.ietf.org/rfc/rfc1256.txt, S. Deering.

25. *RFC 826, An Ethernet Address Resolution Protocol*, http://www.ietf.org/rfc/rfc0826.txt, David C. Plummer.

26. *RFC Sourcebook – Protocols*, http://www.networksorcery.com/enp/Protocol.htm, Network Sourcery, Inc.

27. *RIP & OSPF Redistribution*, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ics/cs001.htm, Cisco online documentation.

28. *Routing & Tunneling Protocol Attacks*, http://www.phenoelit.de/stuff/routing.pdf, Presentation at DEFCON 9, FX / Phenoelit.

29. *SANS Training, Track 4 – Advanced Incident Handling & Hacker Exploits*, SANS Institute.

30. *Securing the Boarder Gateway Protocol*, http://citeseer.nj.nec.com/smith96securing.html, Bradley Smith & J.J. Garcia-Luna-Aceves.

31. *Security for Wide Area Internet Routing*, http://www.acm.org/crossroads/columns/onpatrol/november00.html, Nick Feamster.

32. *Security Issues Affecting Transit Points and Backbone Providers*, http://www.blackhat.com/presentations/bh-usa-99/Batz/blackhat-batz.ppt, Presentation at Black Hat Briefings 1999, Batz.

33. *TCP/IP Illustrated, Volume 1 – The Protocols*, Richard Stevens, Addison Wesley.