

Cisco Applied Mitigation Bulletin: Identifying and Mitigating Exploitation of the Vulnerability in Crypto Library

Document ID: 91832

<http://www.cisco.com/warp/public/707/cisco-amb-20070522-crypto.shtml>

Revision 1.0

For Public Release 2007 May 22 1300 UTC (GMT)

Please provide your [feedback](#) on this document.

[Cisco Response](#)
[Device-Specific Mitigation and Identification](#)
[Additional Information](#)
[Revision History](#)
[Cisco Security Procedures](#)
[Related Information](#)

Cisco Response

This Applied Mitigation Bulletin is a companion document to the PSIRT Security Advisory *Vulnerability in Crypto Library* and provides identification and mitigation techniques that administrators can deploy on Cisco network devices.

Vulnerability Characteristics

There is a vulnerability in a third-party cryptographic library component of certain Cisco products when processing a malformed Abstract Syntax Notation One (ASN.1) object when it processes malformed IKE, GDOI, SSH, HTTPS, SIP-TLS, and TIDP packets. This vulnerability can be exploited remotely without authentication and without end-user interaction. Successful exploitation of this vulnerability may cause the affected device to crash. The following protocols are attack vectors for exploitation.

- IKE protocol using UDP port 500
- [Group Domain of Interpretation \(GDOI\)](#) protocol using UDP port 848
- SSH protocol using TCP port 22
- HTTPS protocol using TCP port 443
- [SIP-TLS protocol](#) using TCP port 5060
- [Threat Information Distribution Protocol \(TIDP\)](#) using TCP port 5354

In the case of IKE and GDOI, this vulnerability is susceptible to exploitation through spoofed attacks. This vulnerability has been assigned CVE name CVE-2006-3894.

Information about vulnerable, unaffected, and fixed software is available in the PSIRT Security Advisory: <http://www.cisco.com/warp/public/707/cisco-sa-20070522-crypto.shtml>

Mitigation Technique Overview

Cisco devices provide several countermeasures for the cryptographic library vulnerability. Administrators are advised to consider many of these protection methods to be general security best practices for infrastructure devices and the traffic that transits the network.

Cisco IOS Software can provide effective means of exploit prevention using the following methods:

- Infrastructure access control lists
- Transit access control lists
- Unicast Reverse Path Forwarding
- IP source guard

Cisco IOS XR can provide effective means of exploit prevention using infrastructure access control lists.

Effective means of exploit prevention can also be provided by Cisco ASA 5500 Series Adaptive Security Appliance, Cisco PIX 500 Series Security Appliance, and the Firewall Services Module (FWSM) for Cisco Catalyst 6500 Series switches and Cisco 7600 Series routers using the following methods:

- Transit access control lists
- Unicast Reverse Path Forwarding

These protection mechanisms filter and drop as well as verify the source IP address of packets that are attempting to exploit the vulnerability described in this document.

The proper deployment and configuration of IP source guard provides the most effective means of anti-spoofing protection against attacks with spoofed source MAC addresses. Deployment as close to all traffic sources as possible provides maximum effectiveness.

Cisco IOS NetFlow can provide visibility into these exploitation attempts using flow records. Cisco IOS Software, Cisco ASA, Cisco PIX security appliances, and FWSM firewalls can provide visibility through syslog messages and the counter values displayed in the output from **show** commands.

Risk Management

Organizations should follow their standard risk evaluation and mitigation process to determine the potential impact of this vulnerability. *Triage* refers to sorting projects and prioritizing efforts that are most likely to be successful. Cisco has provided documents that can help organizations develop a risk-based triage capability for their information security teams. [Risk Triage for Security Vulnerability Announcements](#) and [Risk Triage and Prototyping in Information Security Engagements](#) can help organizations develop repeatable security evaluation and response processes.

Device-Specific Mitigation and Identification



Caution: The effectiveness of any mitigation technique is dependent on specific customer situations such as product mix, network topology, traffic behavior, and organizational mission. As with any configuration change, evaluate the impact of this configuration prior to applying the change.

Specific information on mitigation and identification is available for these devices:

- [Cisco IOS Routers and Switches](#)

- [Cisco IOS XR Routers](#)
- [Cisco IOS NetFlow](#)
- [Cisco ASA, PIX, and FWSM Firewalls](#)

Cisco IOS Routers and Switches

Mitigation: Infrastructure Access Control Lists

In an effort to protect infrastructure devices and minimize the risk, impact, and effectiveness of direct infrastructure attacks, administrators should deploy infrastructure access control lists (iACLs) to perform policy enforcement of traffic sent to infrastructure equipment. Administrators can construct an iACL by explicitly permitting only authorized traffic sent to infrastructure devices in accordance with existing security policies and configurations. For the maximum protection of infrastructure devices, deployed iACLs should be applied in the ingress direction on all interfaces to which an IP address has been configured.

In the following example, the address block 192.168.1.0/24 is the infrastructure address space and the host at 192.168.100.1 is considered a trusted endpoint. The iACL policy denies IKE packets on UDP port 500, GDOI packets on UDP port 848, SSH packets on TCP port 22, HTTPS packets on TCP port 443, SIP-TLS packets on TCP port 5060, and TIDP packets on TCP port 5354 sent to addresses that are part of the infrastructure address space. Care should be taken to allow required traffic for routing and administrative access prior to denying all traffic sent directly to infrastructure devices. Whenever possible, infrastructure address space should be distinct from the address spaced used for user and services segments. Using this addressing methodology will assist with the construction and deployment of iACLs.

Additional information about iACLs is available in [Protecting Your Core: Infrastructure Protection Access Control Lists](#).

```
ip access-list extended infrastructure-acl-policy

!
!-- When applicable, include explicit permit statements for trusted
!-- sources that require access on the vulnerable ports
!

permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 500
permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 848
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 22
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 443
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5060
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5354

!
!-- The following vulnerability-specific access control entries
!-- (ACEs) can aid in identification of attacks
!

deny udp any 192.168.1.0 0.0.0.255 eq 500
deny udp any 192.168.1.0 0.0.0.255 eq 848
deny tcp any 192.168.1.0 0.0.0.255 eq 22
deny tcp any 192.168.1.0 0.0.0.255 eq 443
deny tcp any 192.168.1.0 0.0.0.255 eq 5060
deny tcp any 192.168.1.0 0.0.0.255 eq 5354

!
!-- Explicit deny ACE for traffic sent to addresses configured within
!-- the infrastructure address space
!
```

```

deny ip any 192.168.1.0 0.0.0.255

!
!-- Permit/deny all other Layer 3 and Layer 4 traffic in accordance with
!-- existing security policies and configurations
!
!-- Apply iACL to interface(s) in the ingress direction

interface GigabitEthernet0/0
 ip access-group infrastructure-acl-policy in

```

Mitigation: Anti-Spoof Protections

Unicast Reverse Path Forwarding

The cryptographic library vulnerability described in this document can be exploited by spoofed IP packets specifically through the IKE protocol. Protection mechanisms for anti-spoofing exist through the proper deployment and configuration of Unicast Reverse Path Forwarding (Unicast RPF). Unicast RPF is configured at the interface level and can detect and drop packets that lack a verifiable IP source address. Unicast RPF should not be relied on to provide 100 percent anti-spoofing protection because spoofed packets may enter the network through a Unicast RPF-enabled interface if an appropriate return route to the source IP address exists. Care must be taken to ensure that the appropriate Unicast RPF mode (loose or strict) is configured during the deployment of this feature because it can drop legitimate traffic transiting through the network. For additional information please reference the [Unicast Reverse Path Forwarding Loose Mode Feature Guide](#).

IP Source Guard

IP source guard (IPSG) is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering packets based on the DHCP snooping binding database and manually configured IP source bindings. Administrators can use IPSG to prevent attacks from an attacker who attempts to spoof packets by forging the source IP address and/or the MAC address. When properly deployed and configured, IPSG coupled with strict mode Unicast RPF provides the most effective means of anti-spoofing protection.

Additional information about the deployment and configuration of IPSG is available in [Configuring DHCP Features and IP Source Guard](#).

Identification: Infrastructure Access Control Lists

After the administrator applies the infrastructure ACL to an interface, the **show ip access-lists** command will identify the number of packets that have been filtered in all interfaces that the iACL has been applied to. Administrators should investigate filtered packets to determine whether they are attempts to exploit this vulnerability. Example output for **show ip access-lists infrastructure-acl-policy** follows:

```

router#show ip access-lists infrastructure-acl-policy
Extended IP access list infrastructure-acl-policy
 10 permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq isakmp
 20 permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 848
 30 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 22
 40 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 443
 50 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5060
 60 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5354
 70 deny udp any 192.168.1.0 0.0.0.255 eq isakmp
 80 deny udp any 192.168.1.0 0.0.0.255 eq 848
 90 deny tcp any 192.168.1.0 0.0.0.255 eq 22
100 deny tcp any 192.168.1.0 0.0.0.255 eq 443 (127 matches)
110 deny tcp any 192.168.1.0 0.0.0.255 eq 5060
120 deny tcp any 192.168.1.0 0.0.0.255 eq 5354
130 deny ip any 192.168.1.0 0.0.0.255

```

In the preceding example, the access list *infrastructure-acl-policy* has dropped **127** HTTPS packets on TCP port 443 for ACE sequence ID 100.

Identification: Access List Logging

The **log** or **log-input** ACL option will cause packets that match specific ACEs to be logged. The **log-input** option enables logging of the ingress interface in addition to the packet source and destination IP addresses and ports.

Caution: ACL logging can be very CPU intensive and must be used with extreme caution. The CPU impact of ACL logging is driven by two factors: process switching as a result of packets that match log-enabled ACEs and log generation and transmission. The **ip access-list logging interval interval-in-ms** command can limit the effects of process switching induced by ACL logging. The **logging rate-limit rate-per-second [except loglevel]** command limits the impact of log generation and transmission.

The CPU impact from ACL logging can be addressed in hardware on the Catalyst 6500 Series switches and Cisco 7600 Series routers with Supervisor 720 and Supervisor 32 using optimized ACL logging.

Identification: Anti-Spoof Protection Using Unicast RPF

With Unicast RPF properly deployed and configured throughout the network infrastructure, administrators can use the **show ip interface**, **show cef drop**, and **show ip traffic** commands to identify the number of packets that Unicast RPF has dropped.

Note that the **show command | begin regex** and **show command | section regex [command modifiers](#)** are used in the following examples to minimize the amount of output that administrators will need to parse to view the desired information. The **show command | include regex** is another option.

```
router#show ip interface GigabitEthernet 0/0 | begin verify
  IP verify source reachable-via RX, allow default, allow self-ping
  18 verification drops
  0 suppressed verification drops
router#show ip traffic | section IP statistics
IP statistics:
  Rcvd: 68051015 total, 2397325 local destination
        43999 format errors, 0 checksum errors, 33 bad hop count
        2 unknown protocol, 929 not a gateway
        21 security failures, 190123 bad options, 542768 with options
  Opts: 352227 end, 452 nop, 36 basic security, 1 loose source route
        45 timestamp, 59 extended security, 41 record route
        53 stream ID, 3 strict source route, 40 alert, 45 cipso, 0 ump
        361634 other
  Frags: 0 reassembled, 10008 timeouts, 56866 couldn't reassemble
        0 fragmented, 0 fragments, 0 couldn't fragment
  Bcast: 64666 received, 0 sent
  Mcast: 1589885 received, 2405454 sent
  Sent: 3001564 generated, 65359134 forwarded
  Drop: 4256 encapsulation failed, 0 unresolved, 0 no adjacency
        18 no route, 18 unicast RPF, 0 forced drop
        0 options denied
  Drop: 0 packets with source IP address zero
  Drop: 0 packets with internal loop back IP address
```

In the preceding examples, Unicast RPF has dropped **18** IP packets received on interfaces with Unicast RPF configured due to the inability to verify the source address of the IP packets within the Cisco Express Forwarding Forwarding Information Base.

Cisco IOS XR Routers

Mitigation: Infrastructure Access Control Lists

In an effort to protect infrastructure devices and minimize the risk, impact, and effectiveness of direct infrastructure attacks, administrators should deploy infrastructure ACLs to perform policy enforcement of traffic sent to infrastructure equipment. Administrators can construct an iACL by explicitly permitting only authorized traffic sent to infrastructure devices in accordance with existing security policies and configurations. For the maximum protection of infrastructure devices, deployed iACLs should be applied in the ingress direction on all interfaces to which an IP address has been configured.

In the following example, the address block 192.168.1.0/24 is the infrastructure address space and the host at 192.168.100.1 is considered a trusted endpoint. The iACL policy denies IKE packets on UDP port 500, GDOI packets on UDP port 848, SSH packets on TCP port 22, HTTPS packets on TCP port 443, SIP-TLS packets on TCP port 5060, and TIDP packets on TCP port 5354 sent to addresses that are part of the infrastructure address space. Care should be taken to allow required traffic for routing and administrative access prior to denying all traffic sent directly to infrastructure devices. Whenever possible, infrastructure address space should be distinct from the address spaced used for user and services segments. Using this addressing methodology will assist with the construction and deployment of iACLs.

```
ipv4 access-list infrastructure-acl-policy

!
!-- When applicable, include explicit permit statements for trusted
!-- sources that require access on the vulnerable ports
!

permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 500
permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 848
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 22
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 443
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5060
permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5354

!
!-- The following vulnerability-specific access control entries
!-- (ACEs) can aid in identification of attacks
!

deny udp any 192.168.1.0 0.0.0.255 eq 500
deny udp any 192.168.1.0 0.0.0.255 eq 848
deny tcp any 192.168.1.0 0.0.0.255 eq 22
deny tcp any 192.168.1.0 0.0.0.255 eq 443
deny tcp any 192.168.1.0 0.0.0.255 eq 5060
deny tcp any 192.168.1.0 0.0.0.255 eq 5354

!
!-- Explicit deny ACE for traffic sent to addresses configured within
!-- the infrastructure address space
!

deny ipv4 any 192.168.1.0 0.0.0.255
permit ip any any

!
!-- Permit/deny all other Layer 3 and Layer 4 traffic in accordance with
!-- existing security policies and configurations
!
!-- Apply iACL to interface(s) in the ingress direction
```

```
interface POS0/5/0/3
  ipv4 access-group infrastructure-acl-policy ingress
```

Identification: Infrastructure Access Control Lists

After the administrator applies the infrastructure ACL to an interface, the **show access-lists ipv4 access-list-name hardware ingress location node-id** command will identify the number of packets that have been filtered by the line card. Administrators should investigate filtered packets to determine whether they are attempts to exploit this vulnerability. Example output for **show ip access-lists ipv4 infrastructure-acl-policy** follows:

```
gsr#show access-lists ipv4 infrastructure-acl-policy hardware ingress location 0/5/CPU0
ipv4 access-list infrastructure-acl-policy
 10 permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq isakmp
 20 permit udp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 848
 30 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 22
 40 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 443
 50 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5060
 60 permit tcp host 192.168.100.1 192.168.1.0 0.0.0.255 eq 5354
 70 deny udp any 192.168.1.0 0.0.0.255 eq isakmp
 80 deny udp any 192.168.1.0 0.0.0.255 eq 848
 90 deny tcp any 192.168.1.0 0.0.0.255 eq 22 (5 hw matches)
100 deny tcp any 192.168.1.0 0.0.0.255 eq 443
110 deny tcp any 192.168.1.0 0.0.0.255 eq 5060
120 deny tcp any 192.168.1.0 0.0.0.255 eq 5354
130 deny ipv4 any 192.168.1.0 0.0.0.255 (28 hw matches)
```

In the preceding example, the access list *infrastructure-acl-policy* has dropped **five** SSH packets on TCP port 22 for ACE sequence ID 90.

Additional information about the **show access-lists ipv4** command is available in [Access List Commands on Cisco IOS XR Software](#).

Cisco IOS NetFlow

Identification: Traffic Flow Identification Using NetFlow Records

Administrators can configure Cisco IOS NetFlow on Cisco IOS routers and switches to aid in the identification of traffic flows that may be potential attempts to exploit the vulnerability described in this document. Administrators should investigate flows to determine whether they are attempts to exploit this vulnerability or if they are legitimate traffic flows.

```
router#show ip cache flow
IP packet size distribution (157631359 total packets):
 1-32  64  96 128 160 192 224 256 288 320 352 384 416 448 480
 .011 .572 .054 .008 .009 .010 .005 .013 .000 .001 .004 .001 .003 .002 .006

 512 544 576 1024 1536 2048 2560 3072 3584 4096 4608
 .001 .001 .158 .013 .117 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 4456704 bytes
 14 active, 65522 inactive, 66980401 added
236759230 ager polls, 0 flow alloc failures
Active flows timeout in 1 minutes
Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 402056 bytes
 28 active, 16356 inactive, 8160312 added, 6507913 added to flow
 0 alloc failures, 0 force free
 1 chunk, 11 chunks added
```

last clearing of statistics never

Protocol	Total	Flows	Packets	Bytes	Packets	Active(Sec)	Idle(Sec)
-----	Flows	/Sec	/Flow	/Pkt	/Sec	/Flow	/Flow
TCP-Telnet	11416733	2.6	1	49	3.1	0.0	1.5
TCP-FTP	14591	0.0	5	53	0.0	3.6	11.0
TCP-FTPD	1038	0.0	2167	881	0.5	36.9	1.6
TCP-WWW	207227	0.0	12	713	0.5	4.3	9.2
TCP-SMTP	12	0.0	1	47	0.0	0.0	10.5
TCP-X	731	0.0	1	40	0.0	0.0	1.4
TCP-BGP	13	0.0	1	46	0.0	0.0	10.3
TCP-NNTP	12	0.0	1	47	0.0	0.0	9.7
TCP-Frag	70401	0.0	1	688	0.0	0.0	22.7
TCP-other	49753723	11.5	2	337	29.8	0.1	1.5
UDP-DNS	1916648	0.4	1	56	0.5	0.0	15.4
UDP-NTP	1773812	0.4	1	76	0.4	0.4	15.5
UDP-TFTP	10	0.0	2	57	0.0	6.6	18.6
UDP-other	1370186	0.3	2	161	0.6	0.3	16.4
ICMP	431086	0.1	8	46	0.8	15.1	17.8
IPv6INIP	15	0.0	1	1132	0.0	0.0	15.4
GRE	23980	0.0	19	109	0.1	57.4	1.7
IP-other	2	0.0	2	20	0.0	0.1	15.7
Total:	66980220	15.5	2	309	36.7	0.2	2.7

SrcIf	SrcIPAddress	DstIf	DstIPAddress	Pr	SrcP	DstP	Pkts
Gi0/0	192.168.211.201	Gi0/0	192.168.131.10	11	0A89	01F4	1
Gi0/0	192.168.109.132	Gi0/1	192.168.150.60	06	1A29	BCD4	3
Gi0/0	192.168.226.1	Gi0/1	192.168.202.22	11	007B	007B	1
Gi0/0	192.168.12.189	Gi0/0	192.168.131.10	11	0A8A	01F4	1
Gi0/0	192.168.208.63	Local	192.168.208.20	06	8CAA	0017	52
Gi0/0	192.168.5.19	Gi0/0	192.168.1.10	11	0A86	01F4	1
Gi0/0	192.168.148.30	Gi0/0	192.168.1.10	11	0A88	01F4	1
Gi0/0	192.168.226.1	Gi0/1	192.168.210.12	11	007B	007B	1
Gi0/0	192.168.2.17	Gi0/0	192.168.1.10	11	0A87	01F4	1
Gi0/0	192.168.254.17	Gi0/1	192.168.150.1	01	0000	030D	10
Gi0/0	192.168.226.1	Gi0/1	192.168.160.4	11	007B	007B	1
Gi0/0	192.168.128.56	Gi0/1	192.168.132.44	11	0035	E0E5	2
Gi0/0	192.168.208.63	Gi0/0	192.168.66.197	06	0016	C230	42
Gi0/0	192.168.208.63	Gi0/0	192.168.66.197	06	0016	C0B0	39

In the preceding example, there are multiple flows for IKE on UDP port 500 (hex value 01F4). This traffic is being sent to addresses within the 192.168.1.0/24 address block, which is used for infrastructure devices. The packets in these flows may be spoofed and may indicate an attempt to exploit the vulnerability described in this document. In this example, the source ports are sequential (hex values 0A86, 0A87, and 0A88), which indicate spoofed IP addresses coming from the same host despite the different source IP addresses. These flows should be compared to baseline utilization for IKE traffic sent on UDP port 500 and they should also be investigated to determine whether the flows are sourced from untrusted hosts or networks. To view only the traffic flows for IKE packets on UDP (hex value 11) port 500 (hex value 01F4) or GDOI packets on UDP (hex value 11) port 848 (hex value 0350), the command **show ip cache flow | include SrcIf|_11.*(01F4|0350)** may be used to display these NetFlow records as shown here:

```
router#show ip cache flow | include SrcIf|_11.*(01F4|0350)
```

SrcIf	SrcIPAddress	DstIf	DstIPAddress	Pr	SrcP	DstP	Pkts
Gi0/0	192.168.152.170	Gi0/0	192.168.1.10	11	0766	01F4	1
Gi0/0	192.168.1.232	Gi0/0	192.168.1.10	11	076F	01F4	1
Gi0/0	192.168.38.0	Gi0/0	192.168.1.10	11	0765	01F4	1
Gi0/0	192.168.151.147	Gi0/0	192.168.1.10	11	076B	01F4	1
Gi0/0	192.168.230.192	Gi0/0	192.168.1.10	11	076E	01F4	1
Gi0/0	192.168.109.148	Gi0/0	192.168.1.10	11	076A	01F4	1
Gi0/0	192.168.101.252	Gi0/0	192.168.1.10	11	0769	01F4	1
Gi0/0	192.168.9.107	Gi0/0	192.168.1.10	11	076C	01F4	1
Gi0/0	192.168.231.55	Gi0/0	192.168.1.10	11	0767	01F4	1
Gi0/0	192.168.109.226	Gi0/0	192.168.1.10	11	0768	01F4	1
Gi0/0	192.168.221.86	Gi0/0	192.168.1.10	11	076D	01F4	1

To display flows of SSH packets on TCP (hex value 06) port 22 (hex value 0016), HTTPS packets on TCP (hex value 06) port 443 (hex value 01BB), SIP-TLS packets on TCP (hex value 06) port 5060 (hex value 13C4) or TIDP packets on TCP (hex value 06) port 5354 (hex value 14EA) use the following command:

```
router#show ip cache flow | include SrcIf|_06_.*(0016|01BB|13C4|14EA)
```

The previous regular expression matches on protocol 06 (TCP) followed by zero or more characters (represented by the dot and asterisk metacharacters), followed by *01BB*, *0016*, *13C4*, or *14EA*. This permits matching the flow in either the source or destination ports.

To display all possibly affected flows, the previous commands can be combined as shown in the following example:

```
router#show ip cache flow | include SrcIf|_11_.*(01F4|0350)|_06_.*(0016|01BB|13C4|14EA)
SrcIf      SrcIPAddress  DstIf      DstIPAddress  Pr SrcP DstP  Pkts
Gi0/0      192.168.208.63 Gi0/1      192.168.130.41 06 8B3A 0016  185
Gi0/0      192.168.70.57  Gi0/1      192.168.210.11 06 F04F 01BB  30
Gi0/0      192.168.101.252 Gi0/0      192.168.1.10   11 0769 01F4   1
Gi0/0      192.168.208.63 Gi0/0      192.168.20.143 06 0016 0F6F  94
Gi0/0      192.168.119.165 Gi0/1      192.168.132.44 06 11BE 0016  19
```

Cisco ASA, PIX, and FWSM Firewalls

Mitigation: Transit Access Control Lists

In an effort to protect the network from traffic that enters the network at ingress access points, which may include Internet connection points, partner and supplier connection points, or VPN connection points, administrators should deploy transit ACLs (tACLs) to perform policy enforcement. Administrators can construct a tACL by explicitly permitting only authorized traffic to enter the network at ingress access points or permitting authorized traffic to transit the network in accordance with existing security policies and configurations.

The tACL policy denies unauthorized IKE packets on UDP port 500, GDOI packets on UDP port 848, SSH packets on TCP port 22, HTTPS packets on TCP port 443, SIP-TLS packets on TCP port 5060 or TIDP packets on TCP port 5354 sent to affected devices. In the following example, 192.168.1.0/24 is the network IP address space used by the affected devices and the host at 192.168.100.1 is considered a trusted source that requires access to the affected devices. Care should be taken to allow required traffic for routing and administrative access prior to denying all unauthorized traffic.

Additional information about tACLs is available at [Transit Access Control Lists: Filtering at Your Edge](#).

```
!
!-- Include any explicit permit statements for trusted sources that require access
!-- on the vulnerable port(s)
!

access-list transit-acl-policy extended permit udp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended permit udp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended permit tcp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended permit tcp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended permit tcp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended permit tcp host 192.168.100.1 192.168.1.0 255.255.

!
!-- The following vulnerability-specific access control entries (ACEs) can aid
!-- in identification of attacks
!
```

```

access-list transit-acl-policy extended deny udp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended deny udp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended deny tcp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended deny tcp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended deny tcp host 192.168.100.1 192.168.1.0 255.255.
access-list transit-acl-policy extended deny tcp host 192.168.100.1 192.168.1.0 255.255.

!
!-- Permit/deny all other Layer 3 and Layer 4 traffic in accordance with
!-- existing security policies and configurations
!
!-- Explicit deny for all other IP traffic
!

access-list transit-acl-policy extended deny ip any any

!
!-- Apply tACL to interface(s) in the ingress direction
!

access-group transit-acl-policy in interface outside

```

Mitigation: Anti-Spoof Protection Using Unicast RPF

Attackers can exploit the cryptographic library vulnerability described in this document using spoofed IP packets. Administrators can deploy and configure Unicast RPF as a protection mechanism against anti-spoofing. Unicast RPF is configured at the interface level and can detect and drop packets that lack a verifiable IP source address. Administrators should not rely on Unicast RPF to provide 100 percent anti-spoofing protection because spoofed packets may enter the network through a Unicast RPF-enabled interface if an appropriate return route to the source IP address exists. In an enterprise environment, Unicast RPF might be enabled at the Internet edge and the internal access layer on the user-supporting Layer 3 interfaces.

For additional information about the configuration and use of Unicast RPF, reference the Cisco Security Appliance Command Reference for [ip verify reverse-path](#).

Identification: Transit Access Control Lists

After the tACL has been applied to an interface, administrators can use the **show access-list** command to identify the number of HTTPS packets on TCP port 443, SSH packets on TCP port 22, and IKE packets on UDP port 500 that have been filtered. Administrators should investigate filtered packets to determine whether they are attempts to exploit this vulnerability. Example output for **show access-list transit-acl-policy** follows:

```

Firewall#show access-list transit-acl-policy
access-list transit-acl-policy; 13 elements
access-list transit-acl-policy line 1 extended permit udp host 192.168.100.1 192.168.1.0 2
access-list transit-acl-policy line 2 extended permit udp host 192.168.100.1 192.168.1.0 2
access-list transit-acl-policy line 3 extended permit tcp host 192.168.100.1 192.168.1.0 2
access-list transit-acl-policy line 4 extended permit tcp host 192.168.100.1 192.168.1.0 2
access-list transit-acl-policy line 5 extended permit tcp host 192.168.100.1 192.168.1.0 2
access-list transit-acl-policy line 6 extended permit tcp host 192.168.100.1 192.168.1.0 2
access-list transit-acl-policy line 7 extended deny udp host 192.168.100.1 192.168.1.0 255
access-list transit-acl-policy line 8 extended deny udp host 192.168.100.1 192.168.1.0 255
access-list transit-acl-policy line 9 extended deny tcp host 192.168.100.1 192.168.1.0 255
access-list transit-acl-policy line 10 extended deny tcp host 192.168.100.1 192.168.1.0 25
access-list transit-acl-policy line 11 extended deny tcp host 192.168.100.1 192.168.1.0 25
access-list transit-acl-policy line 12 extended deny tcp host 192.168.100.1 192.168.1.0 25
access-list transit-acl-policy line 13 extended deny ip any any (hitcnt=419) 0xe6ed5ca9

```

In the preceding example, the access list *transit-acl-policy* has dropped **seven** HTTPS packets on TCP port 443 received from an untrusted host or network.

In addition, syslog message 106023 can provide valuable information, which includes the source and destination IP address, the source and destination port numbers, and the IP protocol for the denied packet.

Identification: Firewall Access List Syslog Messages

Firewall syslog message 106023 will be generated for packets denied by an ACE that does not have the **log** keyword present. Additional information about this syslog message is available in [Cisco Security Appliance System Log Message – 106023](#).

Information about configuring syslog for the Cisco ASA 5500 Series Adaptive Security Appliance or the Cisco PIX 500 Series Security Appliance is available in [Configuring Logging on the Cisco Security Appliance](#). Information about configuring syslog on the FWSM for Cisco Catalyst 6500 Series switches and Cisco 7600 Series routers is available in [Configuring Monitoring and Logging on the Cisco FWSM](#).

In the following examples, the **show logging | include regex** command extracts syslog messages from the logging buffer on the firewall. These messages provide additional information about denied packets that could indicate potential attempts to exploit the vulnerability described in this document. It is possible to use different regular expressions with the **include** filtering option to search for specific data in the logged messages. Additional information about regular expression syntax is available in [Using the Command Line Interface](#).

```
firewall# show logging | include 106023
Feb 21 2007 00:15:13: %ASA-4-106023: Deny udp src outside:192.168.2.18/2944 dst inside:192
Feb 21 2007 00:15:13: %ASA-4-106023: Deny udp src outside:192.168.3.200/2945 dst inside:19
```

In the preceding example, the messages logged for the tACL *transit-acl-policy* show potentially spoofed IKE packets on UDP port 500 sent to the address block assigned to the network infrastructure.

Additional information about syslog messages for ASA and PIX security appliances is available in [Cisco Security Appliance System Log Messages](#). Additional information about syslog messages for the FWSM is available in [Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Logging Configuration and System Log Messages](#).

Identification: Anti-Spoof Protection Using Unicast RPF

Firewall syslog message 106021 will be generated for packets denied by Unicast RPF. Additional information about this syslog message is available in [Cisco Security Appliance System Log Message – 106021](#). Information about configuring syslog for the Cisco ASA 5500 Series Adaptive Security Appliance or the Cisco PIX 500 Series Security Appliance is available in [Configuring Logging on the Cisco Security Appliance](#). Information about configuring syslog on the FWSM for Cisco Catalyst 6500 Series switches and Cisco 7600 Series routers is available in [Configuring Monitoring and Logging on the Cisco FWSM](#). In the following example, the **show logging | include regex** command extracts syslog messages from the logging buffer on the firewall. These messages provide additional information about denied packets that could indicate potential attempts to exploit the vulnerability described in this document. It is possible to use different regular expressions with the **include** keyword to search for specific data in the logged messages. Additional information about regular expression syntax is available in [Using the Command Line Interface](#).

```
firewall#show logging | include 106021
May 11 2007 18:09:29: %FWSM-1-106021: Deny icmp reverse path check from 192.168.10.1 to 192
May 11 2007 18:12:15: %FWSM-1-106021: Deny tcp reverse path check from 192.168.10.1 to 192
```

The Cisco ASA **show asp drop** command can also identify the number of packets that Unicast RPF has dropped, as shown in the following example:

```
firewall#show asp drop

Frame drop:
  No valid adjacency                8472
  Reverse-path verify failed      43
  Flow is denied by configured rule  262
  First TCP packet not SYN           34
  TCP RST/FIN out of order           32
  Interface is down                  3

Flow drop:
  Inspection failure                 2
```

In the preceding example, Unicast RPF has dropped a total of **43** IP packets received on all interfaces where it is configured.

For additional information about the configuration and use of Unicast RPF, reference the Cisco Security Appliance Command Reference for [show asp drop](#).

Additional Information

THIS DOCUMENT IS PROVIDED ON AN "AS IS" BASIS AND DOES NOT IMPLY ANY KIND OF GUARANTEE OR WARRANTY, INCLUDING THE WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR USE. YOUR USE OF THE INFORMATION ON THE DOCUMENT OR MATERIALS LINKED FROM THE DOCUMENT IS AT YOUR OWN RISK. CISCO RESERVES THE RIGHT TO CHANGE OR UPDATE THIS DOCUMENT AT ANY TIME.

Revision History

Revision 1.0	2007 May 22	Initial public release
--------------	------------------------	-----------------------------------

Cisco Security Procedures

Complete information on reporting security vulnerabilities in Cisco products, obtaining assistance with security incidents, and registering to receive security information from Cisco, is available on Cisco's worldwide website at http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html. This includes instructions for press inquiries regarding Cisco security notices. All Cisco security advisories are available at <http://www.cisco.com/go/psirt>.

Related Information

- [Applied Mitigation Bulletins](#)
- [Protecting Your Core: Infrastructure Protection Access Control Lists](#)
- [Transit Access Control Lists: Filtering at Your Edge](#)
- [Network Foundation Protection White Papers](#)
- [Cisco IOS NetFlow](#)
- [Cisco Security Appliance System Log Messages, Version 7.2](#)
- [Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Logging Configuration and System Log Messages](#)
- [Understanding ACL on Catalyst 6500 Series Switches](#)
- [Cisco 6.x Intrusion Prevention System](#)

- [Cisco IPS Risk Rating Explained](#)
 - [IPS 6.x Signature Downloads](#)
 - [IPS 5.x Signature Downloads](#)
-

[Contacts & Feedback](#) | [Help](#) | [Site Map](#) © 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: May 22, 2007

Document ID: 91832
