# Transit Access Control Lists: Filtering at Your Edge

## Contents

## Introduction

This document presents guidelines and recommended deployment techniques for filtering transit and edge traffic at your network ingress points. Transit access control lists (ACLs) are used to increase network security by explicitly permitting only required traffic into your network or networks.

## Transit Filters

### Typical Setup

In most edge network environments, such as a typical enterprise network Internet point of presence, ingress filtering should be used to drop unauthorized traffic at the edge of the network. In certain service provider deployments, this form of edge or transit traffic filtering can also be used effectively to limit the flow of transit traffic to and from customers to specific permitted protocols only. This document focuses on an enterprise deployment model.

This example depicts a typical enterprise Internet connectivity design. Two edge routers, IR1 and IR2, provide direct connectivity to the Internet. Behind these two routers, a pair of firewalls (Cisco PIXes in this example) provides stateful inspection capabilities and access to both the internal network and the demilitarized zone (DMZ). The DMZ contains public-facing services such as DNS and web; this is the only network accessible directly from the public Internet. The internal network should never be accessed directly by the Internet, but traffic sourced from the internal network must be able to reach Internet sites.

The edge routers should be configured to provide a first level of security through the use of inbound ACLs. The ACLs allow only specifically permitted traffic to the DMZ and allow return traffic for internal users accessing the Internet. All nonauthorized traffic should be dropped on the ingress interfaces.

### Transit ACL Sections

In general, a transit ACL is composed of four sections.

- Special-use address and anti-spoofing entries that deny illegitimate sources and packets with source addresses that belong within your network from entering the network from an external source
- Explicitly permitted return traffic for internal connections to the Internet
- Explicitly permitted externally sourced traffic destined to protected internal addresses
- Explicit **deny** statement

## How to Develop a Transit ACL

The first step in the development of a transit ACL is to determine the protocols required within your networks. Although every site has specific requirements, certain protocols and applications are widely used and are most often permitted. For instance, if the DMZ segment provides connectivity for a publicly accessible web server, TCP from the Internet to the DMZ server address(es) on port 80 is required. Similarly, internal connections to the Internet require that the ACL permit return established TCP traffic – traffic that has the acknowledgment (ACK) bit set.

### Identify Required Protocols

The development of this list of required protocols can be a daunting task, but there are several techniques that can be used, as needed, in order to help identify required traffic.

- **Review your local security policy / service policy.**Your local site policy should help provide a baseline of permitted and denied services.
- **Review/audit your firewall configuration.**The current firewall configuration should contain explicit **permit** statements for allowed services. In many cases, you can translate this configuration to ACL format and use it to create the bulk of the ACL entries.
- **Review/audit your applications.**The applications hosted on the DMZ and those used internally can help determine filtering requirements. Review the application requirements in order to provide essential details about the filtering design.
- **Use a classification ACL.**A classification ACL is composed of **permit** statements for the various protocols that could be destined to the internal network. (See Appendix A for a list of commonly used protocols and applications.) Use the **show access-list** command to display a count of access control entry (ACE) hits to identify required protocols. Investigate and understand any suspicious or surprising results before you create explicit **permit** statements for unexpected protocols.
- **Use the Netflow switching feature.**Netflow is a switching feature that, if enabled, provides detailed flow information. If Netflow is enabled on your edge routers, the **show ip cache flow**

command gives a list of protocols logged by Netflow. Netflow cannot identify all protocols, so this technique must be used in conjunction with others.

### Identify Invalid Traffic

In addition to direct protection, the transit ACL should also provide a first line of defense against certain types of invalid traffic on the Internet.

- Deny RFC 1918 space.
- Deny packets with a source address that falls under special-use address space, as defined in RFC 3330.
- Apply anti-spoof filters, in accordance with RFC 2827; your address space should never be the source of packets from outside your autonomous system (AS).

Other types of traffic to consider include:

- **External protocols and IP Addresses that need to communicate with the edge router**ICMP from service provider IP AddressesRouting protocolsIPSec VPN, if an edge router is used as the termination
- **Explicitly permitted return traffic for internal connections to the Internet**Specific Internet Control Message Protocol (ICMP) typesOutbound Domain Name System (DNS) query repliesTCP establishedUser Datagram Protocol (UDP) return trafficFTP data connectionsTFTP data connectionsMultimedia connections
- **Explicitly permitted externally sourced traffic destined to protected internal addresses**VPN TrafficInternet Security Association and Key Management Protocol (ISAKMP)Network Address Translation (NAT) TraversalProprietary encapsulationEncapsulating Security Payload (ESP)Authentication Header (AH)HTTP to web serversSecure Socket Layer (SSL) to web serversFTP to FTP serversInbound FTP data connectionsInbound FTP passive (**pasv**) data connectionsSimple Mail Transfer Protocol (SMTP)Other applications and serversInbound DNS queriesInbound DNS zone transfers

### Apply the ACL

The newly constructed ACL should be applied inbound on Internet-facing interfaces of the edge routers. In the example illustrated in the Typical Setup section, the ACL isapplied in on the Internet-facing interfaces on IR1 and IR2.

See the sections on deployment guidelines and deployment example for more details.

## ACL Example

This access list provides a simple yet realistic example of typical entries required in a transit ACL. This basic ACL needs to be customized with local site-specific configuration details.

```
!--- Add anti-spoofing entries.
!--- Deny special-use address sources.
!--- Refer to RFC 3330 for additional special use addresses.

access-list 110 deny ip 127.0.0.0 0.255.255.255 any
```

```
access-list 110 deny ip 192.0.2.0 0.0.0.255 any
access-list 110 deny ip 224.0.0.0 31.255.255.255 any
access-list 110 deny ip host 255.255.255.255 any

!--- The deny statement should not be configured
!--- on Dynamic Host Configuration Protocol (DHCP) relays.

access-list 110 deny ip host 0.0.0.0 any

!--- Filter RFC 1918 space.

access-list 110 deny ip 10.0.0.0 0.255.255.255 any
access-list 110 deny ip 172.16.0.0 0.15.255.255 any
access-list 110 deny ip 192.168.0.0 0.0.255.255 any

!--- Permit Border Gateway Protocol (BGP) to the edge router.

access-list 110 permit tcp host bgp_peer gt 1023 host router_ip eq bgp
access-list 110 permit tcp host bgp_peer eq bgp host router_ip gt 1023

!--- Deny your space as source (as noted in RFC 2827).

access-list 110 deny ip your Internet-routable subnet any

!--- Explicitly permit return traffic.
!--- Allow specific ICMP types.

access-list 110 permit icmp any any echo-reply
access-list 110 permit icmp any any unreachable
access-list 110 permit icmp any any time-exceeded
access-list 110 deny icmp any any

!--- These are outgoing DNS queries.

access-list 110 permit udp any eq 53 host primary DNS server gt 1023

!--- Permit older DNS queries and replies to primary DNS server.

access-list 110 permit udp any eq 53 host primary DNS server eq 53

!--- Permit legitimate business traffic.

access-list 110 permit tcp any Internet-routable subnet established
access-list 110 permit udp any range 1 1023 Internet-routable subnet gt 1023

!--- Allow ftp data connections.

access-list 110 permit tcp any eq 20 Internet-routable subnet gt 1023

!--- Allow tftp data and multimedia connections.

access-list 110 permit udp any gt 1023 Internet-routable subnet gt 1023

!--- Explicitly permit externally sourced traffic.
!--- These are incoming DNS queries.

access-list 110 permit udp any gt 1023 host <primary DNS server> eq 53

!-- These are zone transfer DNS queries to primary DNS server.

access-list 110 permit tcp host secondary DNS server gt 1023 host primary DNS server eq 53

!--- Permit older DNS zone transfers.
```

```
access-list 110 permit tcp host secondary DNS server eq 53 host primary DNS server eq 53

!--- Deny all other DNS traffic.

access-list 110 deny udp any any eq 53
access-list 110 deny tcp any any eq 53

!--- Allow IPSec VPN traffic.

access-list 110 permit udp any host IPSec headend device eq 500
access-list 110 permit udp any host IPSec headend device eq 4500
access-list 110 permit 50 any host IPSec headend device
access-list 110 permit 51 any host IPSec headend device
access-list 110 deny ip any host IPSec headend device

!--- These are Internet-sourced connections to
!--- publicly accessible servers.

access-list 110 permit tcp any host public web server eq 80
access-list 110 permit tcp any host public web server eq 443
access-list 110 permit tcp any host public FTP server eq 21

!--- Data connections to the FTP server are allowed !--- by the permit established ACE.
!--- Allow PASV data connections to the FTP server.

access-list 110 permit tcp any gt 1023 host public FTP server gt 1023
access-list 110 permit tcp any host public SMTP server eq 25

!--- Explicitly deny all other traffic.

access-list 101 deny ip any any
```

**Note**: Please keep these suggestions in mind when you apply the transit ACL.

- The **log** keyword can be used in order to provide additional detail about source and destinations for a given protocol. Although this keyword provides valuable insight into the details of ACL hits, excessive hits to an ACL entry that uses the **log** keyword increase CPU utilization. The performance impact associated with logging varies by platform.
- ICMP unreachable messages are generated for packets that are administratively denied by an ACL. This could impact router and link performance. Consider the use of the **no ip unreachables** command in order to disable IP unreachables on the interface where the transit (edge) ACL is deployed.
- This ACL can be initially deployed with all **permit** statements in order to ensure that business legitimate traffic is not denied. Once business legitimate traffic has been identified and accounted for, the specific **deny** elements can be configured.

## ACLs and Fragmented Packets

ACLs have a **fragments** keyword that enables specialized fragmented packet-handling behavior. In general, noninitial fragments that match the Layer 3 statements (protocol, source address, and destination address)—irrespective of the Layer 4 information in an ACL—are affected by the **permit** or **deny** statement of the matched entry. Note that the use of the **fragments** keyword can force ACLs to either deny or permit noninitial fragments with more granularity.

Filtering fragments adds an additional layer of protection against a denial-of-service (DoS) attack that uses only noninitial fragments (such as FO > 0). The use of a **deny** statement for noninitial fragments at the beginning of the ACL denies all noninitial fragments from accessing the router. Under rare circumstances, a valid session might require fragmentation and therefore be filtered if a **deny fragment** statement exists in the ACL. Conditions that might lead to fragmentation include the use of digital certificates for ISAKMP authentication and the use of IPSec NAT Traversal.

For example, consider the partial ACL shown here.

```
access-list 110 deny tcp any Internet routable subnet fragments access-list 110 deny udp any
Internet routable subnet fragments access-list 110 deny icmp any Internet routable subnet
fragments
<rest of ACL>
```

Adding these entries to the beginning of an ACL denies any noninitial fragment access to the network, while nonfragmented packets or initial fragments pass to the next lines of the ACL unaffected by the **deny fragment** statements. The previous ACL snippet also facilitates classification of the attack since each protocol—UDP, TCP, and ICMP—increments separate counters in the ACL.

Since many attacks rely on flooding with fragmented packets, filtering incoming fragments to the internal network provides an added measure of protection and helps ensure that an attack cannot inject fragments by simply matching layer 3 rules in the transit ACL.

Refer to Access Control Lists and IP Fragments for a detailed discussion of the options.

# Risk Assessment

When you deploy transit traffic protection ACLs, consider two key areas of risk.

- Ensure that the appropriate **permit/deny** statements are in place. For the ACL to be effective, you must permit all required protocols.
- ACL performance varies from platform to platform. Before you deploy ACLs, review the performance characteristics of your hardware.

Cisco recommends that you test this design in the lab prior to deployment.

# Appendices

## Commonly Used Protocols and Applications

### TCP Port Names

This list of TCP port names can be used instead of port numbers when you configure the ACL in Cisco IOS® Software. Refer to the RFC of the current assigned number in order to find a reference to these protocols. Port numbers that correspond to these protocols can also be found by while you configure the ACL by entering a **?** in place of a port number.

| | |
|---|---|
| bgp | kshell |
| chargen | login |
| cmd | lpd |

| | |
|---|---|
| daytime | nntp |
| discard | pim |
| domain | pop2 |
| echo | pop3 |
| exec | smtp |
| finger | sunrpc |
| ftp | syslog |
| ftp-data | tacacstalk |
| gopher | telnet |
| hostname | time |
| ident | uucp |
| irc | whois |
| klogin | www |

### UDP Port Names

This list of UDP port names can be used instead of port numbers when you configure the ACL in Cisco IOS Software. Refer to the RFC of the current assigned number in order to find a reference to these protocols. Port numbers that correspond to these protocols can also be found by while you configure the ACL by entering a **?** in place of a port number.

| | |
|---|---|
| biff | ntp |
| bootpc | pim-auto-rp |
| bootps | rip |
| discard | snmp |
| dnsix | snmptrap |
| domain | sunrpc |
| echo | syslog |
| isakmp | tacacs |
| mobile-ip | talk |
| nameserver | tftp |
| netbios-dgm | time |
| netbios-ns | who |
| netbios-ss | xdmcp |

## Deployment Guidelines

Cisco recommends conservative deployment practices. You must have a clear understanding of required protocols in order to successfully deploy transit ACLs. These guidelines describe a very conservative method for the deployment of protection ACLs that use iterative approach.

1. **Identify protocols used in the network with a classification ACL.** Deploy an ACL that permits all the known protocols that are used in the network. This discovery, or classification, ACL should have a source address of **any** and a destination of an IP address or the entire Internet-routable IP subnet. Configure a last entry that permits **ip any any log** in order to help

identify additional protocols that you need to allow.The objective is to determine all required protocols that are in use on the network. Use logging for analysis in order to determine what else might communicate with the router.Please note that the network is at risk of attack while an ACL that consists of all **permit** statements is in place. Perform the classification process as quickly as possible so that proper access controls can be put into place.

2. **Review identified packets and begin to filter access to the internal network.** Once you have identified and reviewed the packets filtered by the ACL in step 1, update the classification ACL to account for newly identified protocols and IP addresses. Add ACL entries for anti-spoofing. As required, substitute specific **deny** entries for **permit** statements in the classification ACL. You can use the **show access-list** command to monitor specific **deny** entries can be monitored for hit count. This provides information about prohibited network access attempts without having to enable logging on ACL entries. The last line of the ACL should be a **deny ip any any**. Once again, the hit count against this last entry can provide information about prohibited access attempts.
3. **Monitor and update the ACL.**Monitor the completed ACL in order to ensure that newly introduced required protocols are added in a controlled manner. If you monitor the ACL, it also provides information about prohibited network access attempts that could provide information about impending attacks.

## Deployment Example

This example shows a transit ACL that protects a network based on this addressing.

- The ISP router IP address is 10.1.1.1.The edge router Internet-facing IP address is 10.1.1.2.The Internet-routable subnet is 192.168.201.0 255.255.255.0.The VPN headend is 192.168.201.100.The web server is 192.168.201.101.The FTP server is 192.168.201.102.The SMTP server is 192.168.201.103.The primary DNS server is 192.168.201.104.The secondary DNS server is 172.16.201.50.

The transit protection ACL was developed based on this information. The ACL permits eBGP peering to the ISP router, provides anti-spoof filters, allows specific return traffic, allows specific inbound traffic, and explicitly denies all other traffic.

```
no access-list 110

!--- Phase 1 – Add anti-spoofing entries.
!--- Deny special-use address sources.
!--- See RFC 3330 for additional special-use addresses.

access-list 110 deny ip 127.0.0.0 0.255.255.255 any
access-list 110 deny ip 192.0.2.0 0.0.0.255 any
access-list 110 deny ip 224.0.0.0 31.255.255.255 any
access-list 110 deny ip host 255.255.255.255 any

!--- This deny statement should not be configured
!--- on Dynamic Host Configuration Protocol (DHCP) relays.

access-list 110 deny ip host 0.0.0.0 any

!--- Filter RFC 1918 space.

access-list 110 deny ip 10.0.0.0 0.255.255.255 any
access-list 110 deny ip 172.16.0.0 0.15.255.255 any
```

```
access-list 110 deny ip 192.168.0.0 0.0.255.255 any

!--- Permit BGP to the edge router.

access-list 110 permit tcp host 10.1.1.1 gt 1023 host 10.1.1.2 eq bgp
access-list 110 permit tcp host 10.1.1.1 eq bgp host 10.1.1.2 gt 1023

!--- Deny your space as source (as noted in RFC 2827).

access-list 110 deny ip 192.168.201.0 0.0.0.255 any

!--- Phase 2 – Explicitly permit return traffic.
!--- Allow specific ICMP types.

access-list 110 permit icmp any any echo-reply
access-list 110 permit icmp any any unreachable
access-list 110 permit icmp any any time-exceeded
access-list 110 deny icmp any any

!--- These are outgoing DNS queries.

access-list 110 permit udp any eq domain host 192.168.201.104 gt 1023

!--- Permit older DNS queries and replies to primary DNS server.

access-list 110 permit udp any eq domain host 192.168.201.104 eq domain

!--- Permit legitimate business traffic.

access-list 110 permit tcp any 192.168.201.0 0.0.0.255 established
access-list 110 permit udp any range 1 1023 192.168.201.0 0.0.0.255 gt 1023

!--- Allow FTP data connections.

access-list 110 permit tcp any eq ftp-data 192.168.201.0 0.0.0.255 gt 1023

!--- Allow TFTP data and multimedia connections.

access-list 110 permit udp any gt 1023 192.168.201.0 0.0.0.255 gt 1023

!--- Phase 3 – Explicitly permit externally sourced traffic.
!--- These are incoming DNS queries.

access-list 110 permit udp any gt 1023 host 192.168.201.104 eq domain

!--- Zone transfer DNS queries to primary DNS server.

access-list 110 permit tcp host 172.16.201.50 gt 1023 host 192.168.201.104 eq domain

!--- Permit older DNS zone transfers.

access-list 110 permit tcp host 172.16.201.50 eq domain host 192.168.201.104 eq domain

!--- Deny all other DNS traffic.

access-list 110 deny udp any any eq domain
access-list 110 deny tcp any any eq domain

!--- Allow IPSec VPN traffic.

access-list 110 permit udp any host 192.168.201.100 eq isakmp
access-list 110 permit udp any host 192.168.201.100 eq non500-isakmp
access-list 110 permit esp any host 192.168.201.100
access-list 110 permit ahp any host 192.168.201.100
```

```
access-list 110 deny ip any host 192.168.201.100


!--- These are Internet-sourced connections to
!--- publicly accessible servers.

access-list 110 permit tcp any host 192.168.201.101 eq www
access-list 110 permit tcp any host 192.168.201.101 eq 443
access-list 110 permit tcp any host 192.168.201.102 eq ftp



!--- Data connections to the FTP server are allowed
!--- by the permit established ACE in Phase 3.
!--- Allow PASV data connections to the FTP server.

access-list 110 permit tcp any gt 1023 host 192.168.201.102 gt 1023
access-list 110 permit tcp any host 192.168.201.103 eq smtp

!--- Phase 4 – Add explicit deny statement.

access-list 110 deny ip any any

Edge-router(config)#interface serial 2/0
Edge-router(config-if)#ip access-group 110 in
```

## Related Information

- Requests for Comments (RFCs)
- Technical Support & Documentation - Cisco Systems