Standard and Extended Access Lists Cheat Sheet for Cisco Beginners

This is a generic cheat sheet and not for a specific use case.

What are Access Lists?

Access Control Lists (ACLs) filter network traffic based on criteria like source/destination IP addresses, protocols, and ports. They permit or deny traffic.

ACL Types

Standard ACLs (1-99, 1300-1999)

- Filter based on **source IP address only**
- Apply **close to destination**
- Less granular control

Extended ACLs (100-199, 2000-2699)

- Filter based on **source IP, destination IP, protocol, and ports**
- Apply **close to source**
- More granular control

Standard ACL Configuration

Numbered Standard ACL

```
Router(config)# access-list [1-99] [permit|deny] [source] [wildcard]
```

Examples

```
Router(config)# access-list 10 permit 192.168.1.0 0.0.0.255
Router(config)# access-list 10 deny 192.168.2.10 0.0.0.0
Router(config)# access-list 10 permit any
```

Named Standard ACL

```
Router(config)# ip access-list standard [name]
Router(config-std-nacl)# [permit|deny] [source] [wildcard]
```

Example

```
Router(config)# ip access-list standard BLOCK_SALES
Router(config-std-nacl)# deny 192.168.10.0 0.0.0.255
Router(config-std-nacl)# permit any
```

Extended ACL Configuration

Numbered Extended ACL

```
Router(config)# access-list [100-199] [permit|deny] [protocol] [source] [destination] [operator port]
```

Examples

```
Router(config)# access-list 100 permit tcp 192.168.1.0 0.0.0.255 any eq 80
Router(config)# access-list 100 deny tcp any 192.168.2.0 0.0.0.255 eq 22
Router(config)# access-list 100 permit ip any any
```

Named Extended ACL

```
Router(config)# ip access-list extended [name]
Router(config-ext-nacl)# [permit|deny] [protocol] [source] [destination] [operator port]
```

Example

```
Router(config)# ip access-list extended WEB_FILTER
Router(config-ext-nacl)# permit tcp any any eq 80
Router(config-ext-nacl)# permit tcp any any eq 443
Router(config-ext-nacl)# deny ip any any
```

Apply ACLs to Interfaces

Apply to Interface

```
Router(config-if)# ip access-group [acl-number|name] [in|out]
```

Examples

```
Router(config-if)# ip access-group 10 in
Router(config-if)# ip access-group 100 out
Router(config-if)# ip access-group BLOCK_SALES in
```

Common Wildcard Masks

Wildcard Mask Examples

- **Single host**: 0.0.0.0
- **Class C network**: 0.0.0.255
- **Class B network**: 0.0.255.255
- **Any host**: 255.255.255.255 (or use "any")

Host and Any Keywords

```
Router(config)# access-list 10 permit host 192.168.1.10
Router(config)# access-list 10 permit any
```

Protocol and Port Examples

Common Protocols

- **ip** = Any IP traffic
- **tcp** = TCP traffic
- **udp** = UDP traffic
- **icmp** = ICMP traffic

Common Ports

- **eq 80** = HTTP
- **eq 443** = HTTPS
- **eq 22** = SSH
- **eq 23** = Telnet
- **eq 53** = DNS

Port Operators

- **eq** = Equal to
- **neq** = Not equal to
- **gt** = Greater than
- **lt** = Less than

Essential Show Commands

View ACLs

```
Router# show access-lists
Router# show access-lists [number|name]
```

View Applied ACLs

```
Router# show ip interface [interface]
Router# show running-config | include access
```

Complete ACL Examples

Standard ACL Example

```
! Block 192.168.10.0 network, allow everything else
access-list 50 deny 192.168.10.0 0.0.0.255
access-list 50 permit any

! Apply to interface
interface gigabit0/0
 ip access-group 50 in
```

Extended ACL Example

```
! Allow HTTP/HTTPS, block everything else
access-list 110 permit tcp any any eq 80
access-list 110 permit tcp any any eq 443
access-list 110 deny ip any any

! Apply to interface
interface gigabit0/1
 ip access-group 110 out
```

Named ACL Example

```
! Create named extended ACL
ip access-list extended OFFICE_FILTER
 permit tcp 192.168.1.0 0.0.0.255 any eq 80
 permit tcp 192.168.1.0 0.0.0.255 any eq 443
 permit icmp any any
 deny ip any any

! Apply to interface
interface gigabit0/0
 ip access-group OFFICE_FILTER in
```

Removing ACLs

Remove ACL

```
Router(config)# no access-list [number]
Router(config)# no ip access-list [standard|extended] [name]
```

Remove from Interface

```
Router(config-if)# no ip access-group [number|name] [in|out]
```

Key Points

- **Implicit deny** at end of every ACL
- **Standard ACLs** close to destination
- **Extended ACLs** close to source
- **Top-down processing** - order matters
- **Wildcard masks** are inverse of subnet masks
- **Always permit what you want first**

Remember: Every ACL ends with an implicit "deny any" - always include permit statements for allowed traffic!