Title:
Cisco CCNA Certification Exam Tutorial: Access List Details You Must Know!

Word Count:
385

Summary:
Access lists are packed with details and pitfalls, and you've got to notice both to pass the CCNA exam. Learn the basics and get certified with Chris Bryant, CCIE #12933.

Keywords:
cisco, ccna, certification, exam, pass, access, list, order, host, lines, order, acl, deny, implicit

Article Body:
To pass the CCNA exam, you have to be able to write and troubleshoot access lists. As you climb the ladder toward the CCNP and CCIE, you'll see more and more uses for ACLs. Therefore, you had better know the basics!

The use of "host" and "any" confuses some newcomers to ACLs, so let's take a look at that first.

It is acceptable to configure a wildcard mask of all ones or all zeroes. A wildcard mask of 0.0.0.0 means the address specified in the ACL line must be matched exactly a wildcard mask of 255.255.255.255 means that all addresses will match the line.

Wildcard masks have the option of using the word host to represent a wildcard mask of 0.0.0.0. Consider a configuration where only packets from IP source 10.1.1.1 should be allowed and all other packets denied. The following ACLs both do that.

```
R3#conf t

R3(config)#access-list 6 permit 10.1.1.1 0.0.0.0

R3(config)#conf t

R3(config)#access-list 7 permit host 10.1.1.1
```

The keyword any can be used to represent a wildcard mask of 255.255.255.255.

```
R3(config)#access-list 15 permit any
```

Another often overlooked detail is the order of the lines in an ACL. Even in a two- or three-line ACL, the order of the lines in an ACL is vital.

Consider a situation where packets sourced from 172.18.18.0 /24 will be denied, but all others will be permitted. The following ACL would do that.

```
R3#conf t

R3(config)#access-list 15 deny 172.18.18.0 0.0.0.255

R3(config)#access-list 15 permit any
```

The previous example also illustrates the importance of configuring the ACL with the lines in the correct order to get the desired results. What would be the result if the lines were reversed?

```
R3#conf t

R3(config)#access-list 15 permit any

R3(config)#access-list 15 deny 172.18.18.0 0.0.0.255
```

If the lines were reversed, traffic from 172.18.18.0 /24 would be matched against the first line of the ACL. The first line is “permit any", meaning all traffic is permitted. The traffic from 172.18.18.0/24 matches that line, the traffic is permitted, and the ACL stops running. The statement denying the traffic from 172.18.18.0 is never run.

The key to writing and troubleshoot access lists is to take just an extra moment to read it over and make sure it's going to do what you intend it to do. It's better to realize your mistake on paper instead of once the ACL's been applied to an interface!