Experiment 7

Chinmay Parikh A059

Questions

1. Explain following types of access control list?
2. Explain basic working of standard ACL?
3. What is wild card mask? How it is useful in ACL? Explain with an example.

Answers

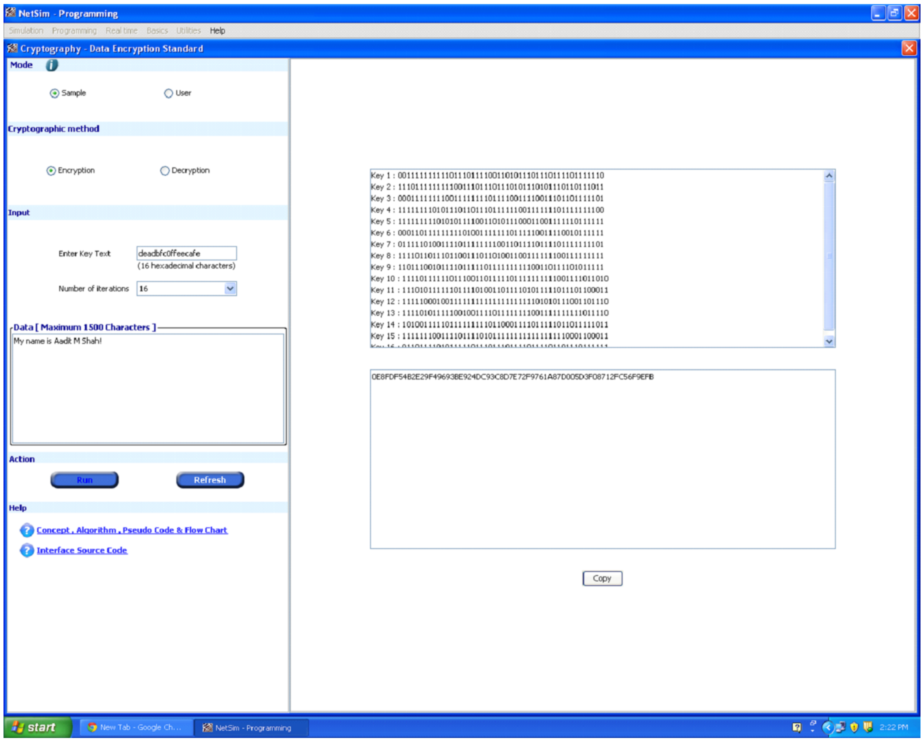
1. - Standard ACL ,This type of AL is the simplest one since it only filters based on source IP addresses. In other words, this AL can be used only when you need to permit or deny traffic from a specific host IP address or a specific source network.
   - Extended ACL ,This type of AL is the most preferred one and the most advanced as well. Using this type of AL you can filter traffic based on: Source IP address Destination IP address Protocol (TCP, UDP) Port Numbers (Ftp 21, Telnet 23, etc.) Supplementary parameters
   - Reflexive ACL, Reflexive access lists allow IP packets to be filtered based on upper-layer session information. Reflexive access lists can be used to permit IP traffic for sessions originating from within your network but to deny IP traffic for sessions originating from outside your network. This is accomplished by reflexive filtering, a kind of session filtering. Reflexive access lists can be defined with extended named IP access lists only. Reflexive access lists cannot be defined with numbered or standard named IP access lists or with other protocol access lists. Reflexive access lists can be used in conjunction with other standard access lists and static extended access lists.
   - Dynamic ACL A dynamic access list is an access list that allows temporary access after a user has authenticated with the router. A dynamic access list could be created giving Cisco complete privileges for a predetermined amount of time. After a configured time limit expires, the session is closed and traffic is again denied. This form of access list is also referred to as-lock-and-key security.
   - Time Based ACL Time-bases ACLs are Access Lists that enable you to restrict or allow resources based on time periods. For example you as a network administrator are asked to restrict web browsing to some particular servers during working hours.

- Named ACL All access lists must be identified by a name or a number. Named access lists are more convenient than numbered access lists because you can specify a meaningful name that is easier to remember and associate with a task. You can reorder statements in or add statements to a named access list. Named access lists support the following features that are not supported by numbered access lists:
  - IP options filtering
  - Noncontiguous ports
  - TCP flag filtering -Deleting of entries with the-no-permit-or-no-deny-command

2. Standard access lists match packets by examining the source IP address field in the packet's IP header. Any bit positions in the 32-bit source IP address can be compared to the access list statements. However, the matching is flexible and does not consider the subnet mask in use. Access lists use the inverse mask, sometimes called the wildcard mask or I-mask. This mask is named because it inverts the meaning of the bits. In a normal mask, ones mean "must match," while zeroes mean "may vary." For example, for two hosts to be on the same Class C network, the first 24 bits of their address must match, while the last 8 may vary. Inverse masks swap the rules so that zeroes mean "must match" and ones mean "may vary." The easy way to calculate the inverse mask is when you already know the normal mask is to subtract from all ones. The table that follows shows an example. The normal mask is subtracted, column by column, from the all-ones mask to determine the inverse mask. All Ones 255 255 255 255 Normal Mask 255 255 240 0 Inverse Mask 0 0 15 255 The command for configuring a standard access list is as follows: Router(config)# access-list {1-99} {permit | deny} source-addr [source-mask]

3. A wildcard mask is a mask of bits that indicates which parts of an IP address are available for examination. A wildcard mask is a sequence of numbers that streamlines packet routing within the subnets of a proprietary network. Wildcard masks are commonly used with OSPF router protocols and in access control lists for Cisco routers. In the Cisco IOS, they are used in several places, for example: To indicate the size of a network or subnet for some routing protocols, such as OSPF. To indicate what IP addresses should be permitted or denied in access control lists (ACLs). At a simplistic level a wildcard mask can be thought of as an inverted subnet mask. For example, a subnet mask of 255.255.255.0 (binary equivalent = 11111111.11111111.11111111.00000000) inverts to a wildcard mask of 0.0.0.255. A wild card mask is a matching rule [2] The rule for a wildcard mask is: 0 means that the equivalent bit must match 1 means that the equivalent bit does not matter Wildcard masks are used in situations where subnet masks may not apply. For example, when two affected hosts fall in different subnets, the use of a wildcard mask will group them together.

Output



Mode: ● Sample ○ User

Cryptographic method: ● Encryption ○ Decryption

Enter Key Text: deadbfc0ffeecafe
(16 hexadecimal characters)

Number of iterations: 16

Data [Maximum 1500 Characters]

My name is Aadit M Shah!

```
Key 1 : 001111111111011101111001101011101110111101111110
Key 2 : 111011111111100111011101110101110101110110111011
Key 3 : 000111111111001111111101111001111001110110111101
Key 4 : 111111110101110110111011111100111111101111111100
Key 5 : 111111111010101111001101011100011001111110111111
Key 6 : 000110111111111010011111110111110011110010111111
Key 7 : 011111010011110111111110011011110111101111111101
Key 8 : 111101101110110011101101001100111111100111111111
Key 9 : 110111001011110111110111111111100110111101011111
Key 10 : 111101111110111001101111101111111111001111011010
Key 11 : 111010111111011110100110111101011111011101100011
Key 12 : 111110001001111111111111111111010101110011011110
Key 13 : 111101011111001001111011111111001111111111011110
Key 14 : 101001111101111111110110001111011111011011111011
Key 15 : 111111100111101111010111111111111111110001100011
Key 16 : 011011110101111101110111011110111101101110111111
```

```
0E8FDF54B2E29F49693BE924DC93C8D7E72F9761A87D005D3F08712FC56F9EFB
```