



CHƯƠNG 6: GIAO THỨC QUẢN LÝ MẠNG

MẠNG MÁY TÍNH NÂNG CAO



SNMP

Giao thức Quản lý mạng



Nội dung



- ❖ Các giao thức quản lý mạng
- ❖ Giao thức SNMP

3

Các thành phần của kiến trúc QLM



- ❖ Tổ chức hệ thống QLM
 - Mô hình tổ chức
- ❖ Chức năng của hệ thống QLM
 - Mô hình chức năng
- ❖ Cơ sở thông tin quản lý
 - Mô hình thông tin
- ❖ Giao thức QLM
 - Mô hình truyền thông

4

Giao thức QLM



- ❖ Là cơ chế trao đổi thông tin giữa Management system và managed system
- ❖ Cung cấp các chức năng:
 - Cấu hình (configuration)
 - Giám sát (monitoring)
 - Cảnh báo (notifying)
 - Kiểm tra và xử lý sự cố (testing and troubleshooting)

5

Các giao thức QLM TCP/IP



- ❖ SNMP
- ❖ NETCONF
- ❖ SYSLOG
- ❖ IPFIX

6

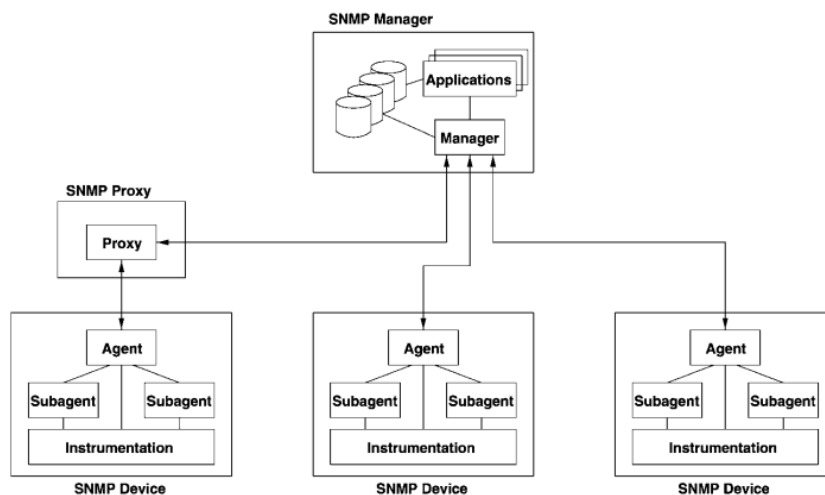
SNMP



- ❖ Xây dựng từ những năm 70 và chuẩn hóa bởi IETF vào năm 1990, 1991 (cùng với SMI và MIB-2)
- ❖ Mục tiêu: đơn giản hóa các chức năng QLM, đặc biệt trên agent.
- ❖ Phiên bản mới nhất (SNMPv3) chuẩn hóa năm 2002.

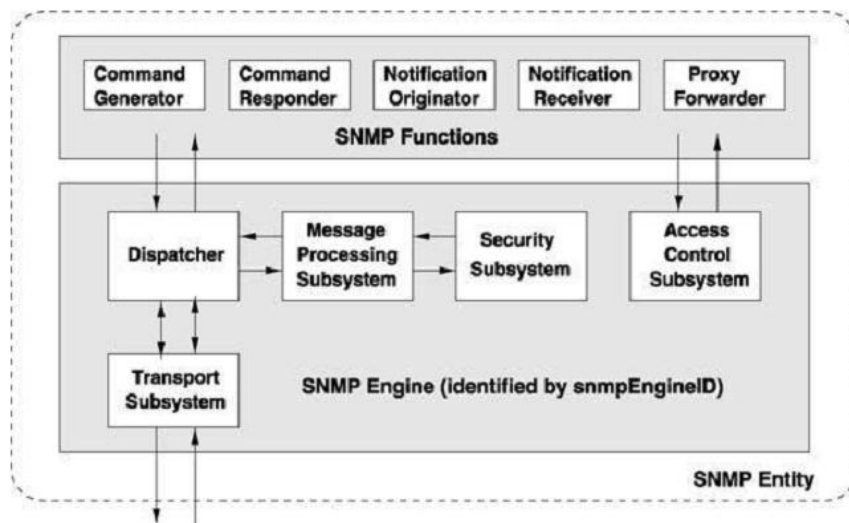
7

Mô hình triển khai SNMP



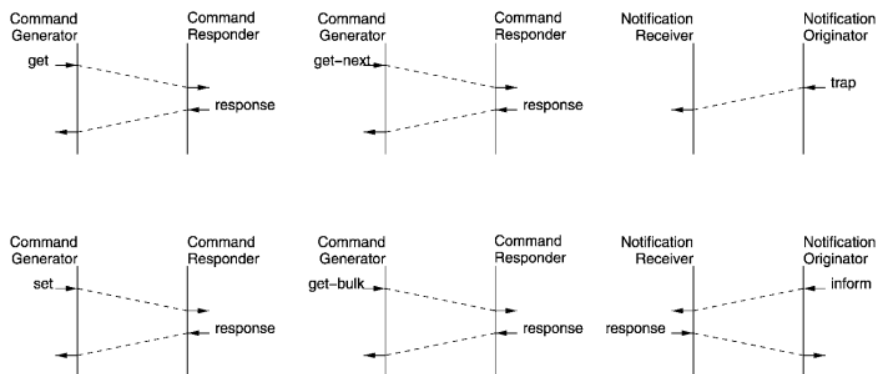
8

Kiến trúc SNMP



9

Cơ chế hoạt động của SNMP



10

Nhận xét



- ❖ **SNMP là giao thức QLM được sử dụng nhiều nhất hiện nay.**
- ❖ **SNMP thích hợp cho các ứng dụng: giám sát thiết bị mạng với số lượng lớn, nhận thông tin cảnh báo từ các thiết bị mạng**
- ❖ **Không thích hợp cho các chức năng cấu hình, điều khiển từ xa.**
- ❖ **Là giao thức nhị phân, khó xử lý thông tin.**
- ❖ **Các ứng dụng kiểm tra và xử lý sự cố đang được phát triển dựa trên SNMP**

11

SNMP



- ❖ **3 phiên bản**
 - SNMP v1: 1990
 - SNMP v2: 1996
 - SNMP v3: 2003
- ❖ **Sử dụng giao thức vận chuyển UDP:**
 - Port 161: polling
 - Port 162: trapping
- ❖ **Là giao thức ứng dụng kiểu nhị phân**

12

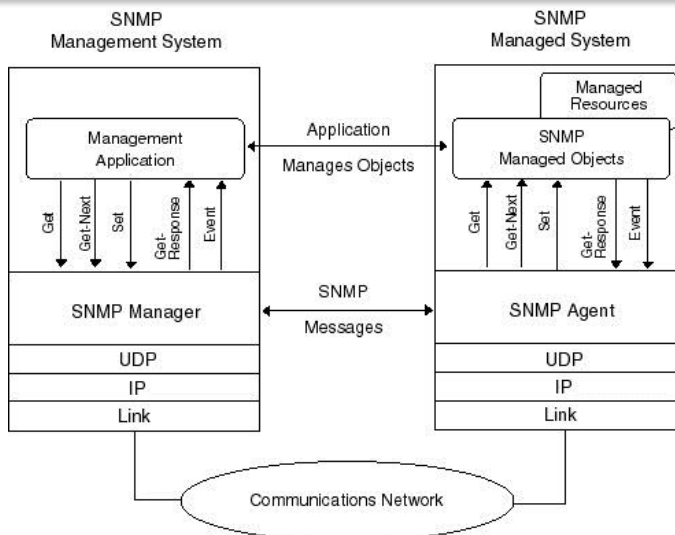
SNMP trong mô hình TCP/IP



OSI Layer	SNMP - Related Function	ARPA Layer
Application	Management Application (SNMP PDU)	Process / Application
Presentation	Structure of Management Information (ASN.1 & BER Encoding)	
Session	Authentication (SNMP Header)	
Transport	User Datagram Protocol (UDP)	Host-to-Host
Network	Internet Protocol (IP)	Internet
Data Link	LAN or WAN Interface Protocol	Network
Physical		Interface

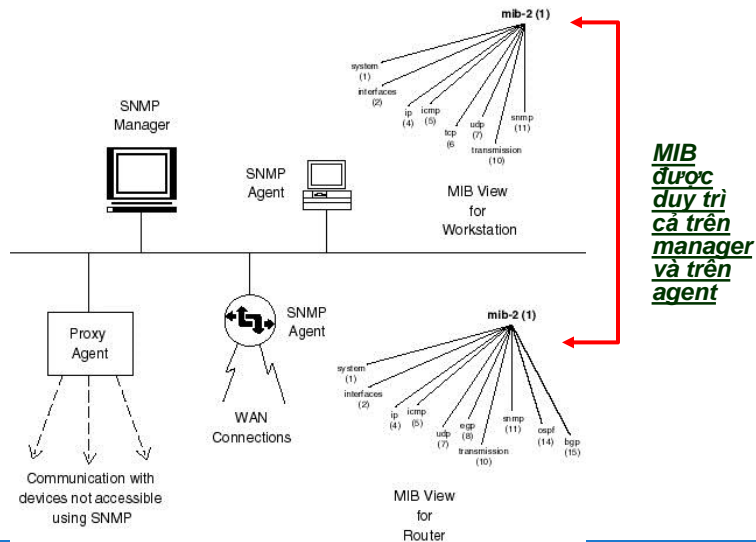
13

Kiến trúc SNMP



14

SNMP trong thực tế



15

Hai chế độ hoạt động của SNMP

❖ Polling:

- Management System gửi yêu cầu truy xuất thông tin đến managed System (port 161)

❖ Trapping:

- Managed System chủ động gửi thông tin cảnh báo cho management system (port 162)

16

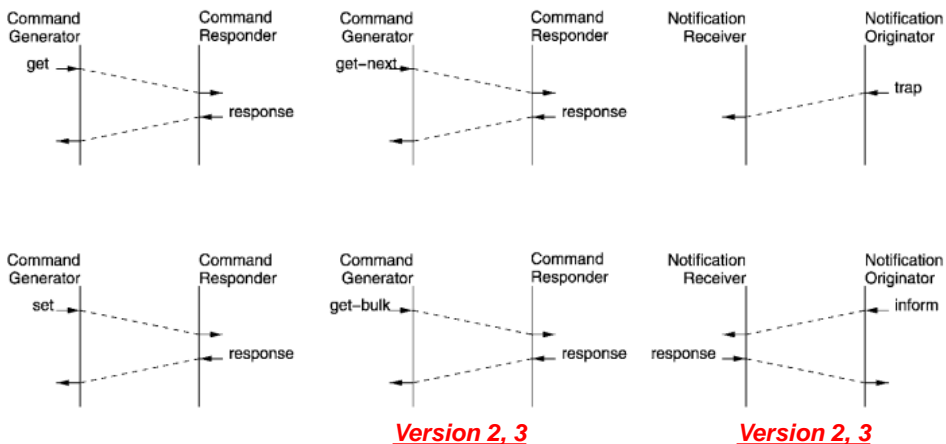
SNMP community string



- ❖ Là chuỗi nhận dạng user (management entity).
- ❖ Có vai trò như một username+password
- ❖ Chức năng: xác định quyền truy xuất của thực thể đối với thông tin quản lý
 - Read-only
 - Read-wite
- ❖ Community string được gọi kèm theo các bản tin request để agent xác thực

17

Các lệnh của SNMP



18

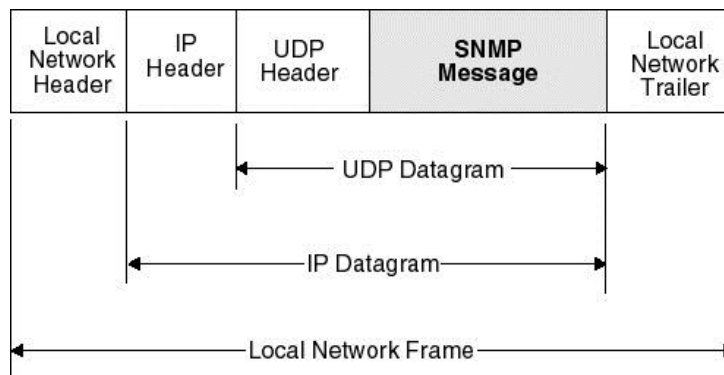
Sử dụng Net-SNMP



- ❖ Net-SNMP là một Management system đơn giản
- ❖ Có thể dùng để truy xuất các thông tin quản lý trên thiết bị có hỗ trợ của SNMP.
 - Tham khảo net-snmp tutorials

19

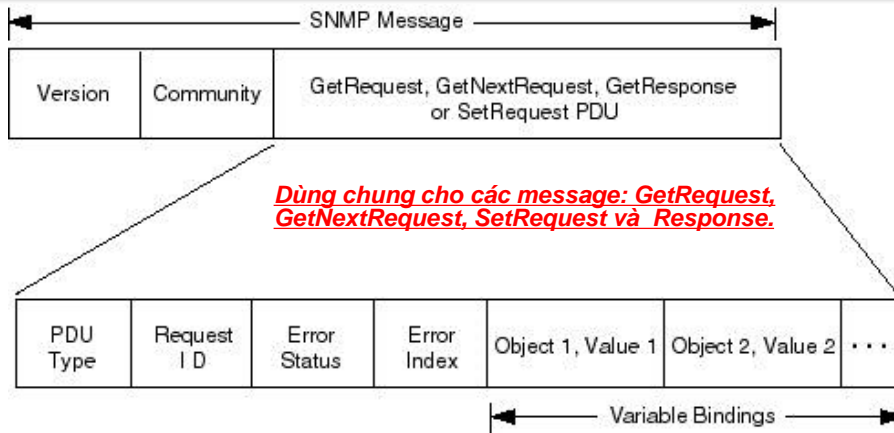
Đóng gói SNMP message



SNMP message được gói trong UDP-IP-Link header

20

Cấu trúc SNMP message



SNMP message = Version + Community string + SNMP PDU

21

PDU type

<u>PDU type</u>	<u>PDU</u>
<u>0</u>	<u>GetRequest</u>
<u>1</u>	<u>GetNextRequest</u>
<u>2</u>	<u>Response</u>
<u>3</u>	<u>SetRequest</u>
<u>4</u>	<u>Trap (obsolete)</u>
<u>5</u>	<u>GetBulkRequest</u>
<u>6</u>	<u>InformRequest</u>
<u>7</u>	<u>Trap (SNMPv2)</u>
<u>8</u>	<u>Report</u>

22

Request ID



❖ Một số nguyên, dùng để nhận dạng request PDU, nhằm mục đích liên kết giữa Request và Response tương ứng.

- Agent có thể nhận cùng lúc nhiều request, mỗi request có một ID khác nhau. Bản tin trả lời (response) của agent sẽ chứa request ID trùng với ID trong bản tin request.

23

Error status & Error index



❖ Error status: số nguyên xác định trạng thái lỗi.

<u>Giá trị</u>	<u>Tên lỗi</u>	<u>Ý nghĩa</u>
0	noError	Không có lỗi
1	tooBig	Kích thước thông tin trả lời vượt quá kích thước PDU
2	noSuchName	Đối tượng không tồn tại
3	badValue	Giá trị không phù hợp
4	readOnly	Không được phép ghi
5	genErr	Lỗi không xác định

- Error index: số nguyên xác định đối tượng (OID) gây lỗi

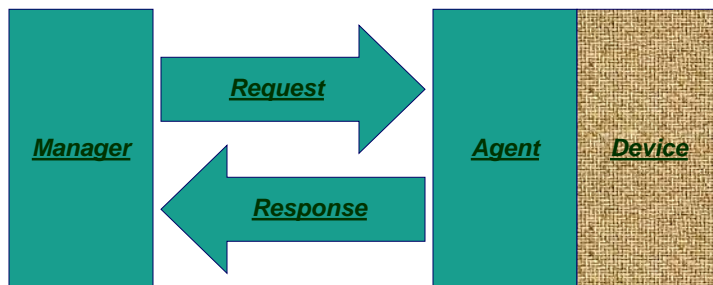
24

Variable bindings

- ❖ Chứa danh sách OID và giá trị tương ứng
- ❖ Mỗi SNMP PDU có thể chứa nhiều variable binding

25

Cơ chế hoạt động của SNMP



26

Thao tác GetRequest



- ❖ Chức năng: yêu cầu lấy giá trị của một hoặc nhiều đối tượng MIB trên SNMP agent.

■ Cấu trúc PDU:

<u>PDU type</u> 0	<u>Request ID</u> x	<u>ErrorStatus</u> 0	<u>ErrorIndex</u> 0	<u>OID 1</u> Null	<u>OID 2</u> Null	<u>OID n</u> Null
----------------------	------------------------	-------------------------	------------------------	----------------------	----------------------	------	----------------------

27

Thao tác GetRequest



■ Câu lệnh net-snmp:

snmpget -v 2c -c public localhost system.sysContact.0

■ Kết quả bắt gói (bỏ IP header)

User Datagram Protocol:

Src Port: 34066 (34066)

Dst Port: snmp (161)

Length: 51

Simple Network Management Protocol Version: 1 (0)

Community: public

PDU type: GET (0)

Request Id: 0x20a71b4c

Error Status: NO ERROR (0) Error Index: 0

Object identifier 1: 1.3.6.1.2.1.1.4.0 (SNMPv2-MIB::sysContact.0)

Value: NULL

28

Thao tác Response

- ❖ Chức năng: trả lại kết quả của thao tác request

- Cấu trúc PDU:

<u>PDU type</u>	<u>Request ID</u>	<u>ErrorStatus</u>	<u>ErrorIndex</u>	<u>OID 1</u>	<u>OID 2</u>	<u>OID n</u>
<u>2</u>	<u>x</u>	<u>0</u>	<u>0</u>	<u>Value 1</u>	<u>Value 2</u>		<u>Value n</u>

- Kết quả bắt gói (bỏ IP header)

```
User Datagram Protocol
  Src Port: snmp (161),
  Dst Port: 34066 (34066)
  Length: 110
Simple Network Management Protocol Version: 1 (0)
  Community: public
  PDU type: RESPONSE (2)
  Request Id: 0x20a71b4c
  Error Status: NO ERROR (0)      Error Index: 0
  Object identifier 1: 1.3.6.1.2.1.1.4.0 (SNMPv2-
MIB::sysContact.0)
  Value: STRING: Root <root@localhost>
```

29

Thao tác GetNextRequest

- ❖ Chức năng: yêu cầu lấy giá trị của một hoặc nhiều đối tượng MIB kế tiếp của đối tượng hiện hành trên SNMP agent.

- Cấu trúc PDU:

<u>PDU type</u>	<u>Request ID</u>	<u>ErrorStatus</u>	<u>ErrorIndex</u>	<u>OID 1</u>	<u>OID 2</u>	<u>OID n</u>
<u>1</u>	<u>x</u>	<u>0</u>	<u>0</u>	<u>Null</u>	<u>Null</u>		<u>Null</u>

- Thao tác GetNextRequest được trả lời bằng Response (PDU type = 2)

30

Thao tác GetNextRequest



■ Câu lệnh net-snmp:

snmpgetnext -v 2c -c public localhost system.sysContact.0

■ Kết quả bắt gói (bỏ IP header)

User Datagram Protocol:

Src Port: 34069 (34069),

Dst Port: snmp (161)

Length: 51

Simple Network Management Protocol Version: 1 (0)

Community: public

PDU type: GET-NEXT (1)

Request Id: 0x00ec480a

Error Status: NO ERROR (0) Error Index: 0

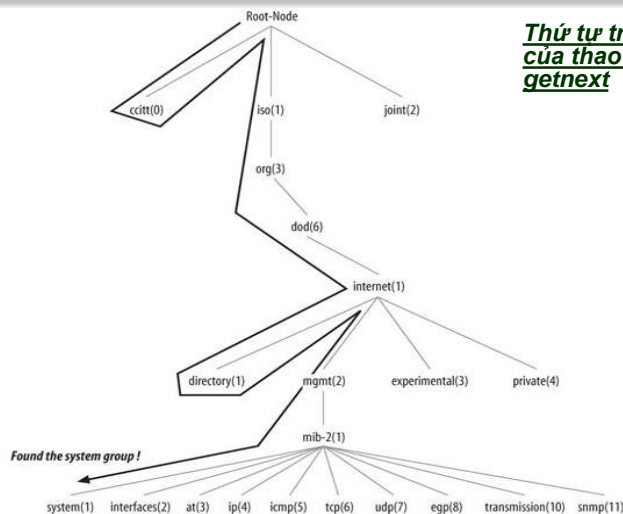
Object identifier 1: 1.3.6.1.2.1.1.1.0 (SNMPv2-

MIB::sysDescr.0)

Value: NULL

31

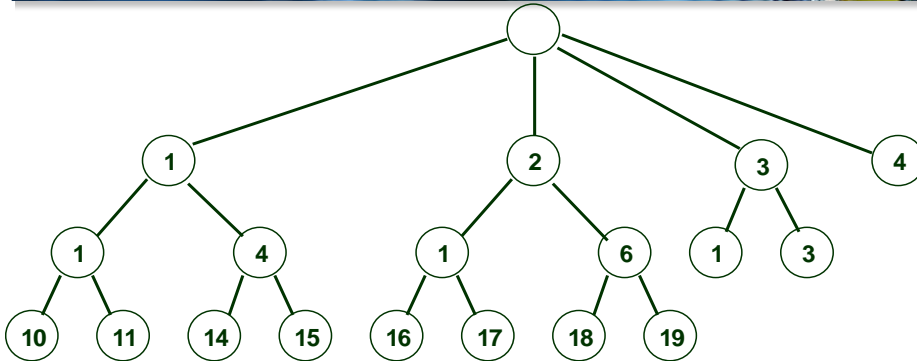
Thao tác GetNextRequest



**Thứ tự truy xuất
của thao tác
getnext**

32

Thao tác GetNextRequest



Thứ tự duyệt cây (lexicographic order):

1 → 1.1 → 1.1.10 → 1.1.11 → 1.4 → 1.4.14 → 1.4.15 → 2
 → 2.1 → 2.1.16 → 2.1.17 → 2.6 → 2.6.18 → 2.6.19 → 3 →
 3.1 → 3.3 → 4

33

Thao tác GetBulkRequest



- ❖ Chức năng: yêu cầu lấy giá trị của một nhóm đối tượng MIB liên nhau trên SNMP agent.

■ Cấu trúc PDU:

PDU type 5	Request ID x	Non-repeater 0	Max repetitions 2	OID 1 Null	OID 2 Null	OID n Null
---------------	-----------------	-------------------	----------------------	---------------	---------------	------	---------------

- Thao tác GetBulkRequest được trả lời bằng Response (type = 2)

34

Thao tác GetBulkRequest



❖ Số đối tượng trả về từ thao tác getBulk:

- N=Non-repeater
- M=Max-repetitions
- R= variable bindings – Non-repeater
- =>return object = N + M*R

❖ Câu lệnh net-snmp:

```
snmpbulkget -v2c -c public -Cn1 -Cr3 localhost sysDescr ifInOctets  
ifOutOctets
```

-Cn1: Non-repeater = 1

-Cr3: Max-repetitions=3

Thực hiện 1 (Cn1) thao tác getNext với 1 OID (sysDescr) và 3 (Cr3) thao tác getNext ứng với mỗi OID còn lại (ifInOctets và ifOutOctets)

35

Thao tác GetBulkRequest



```
snmpbulkget -v2c -c public -Cn1 -Cr2 127.0.0.1 sysDescr sysContact
```

Kết quả bắt gói:

User Datagram Protocol:

Src Port: 34193 (34193)

Dst Port: snmp (161)

Length: 63

Simple Network Management Protocol Version: 2C (1)

Community: public

PDU type: GETBULK (5)

Request Id: 0x0f15c607

Non-repeaters: 1 Max repetitions: 2

Object identifier 1: 1.3.6.1.2.1.1.1 (SNMPv2-MIB::sysDescr)

Value: NULL Object identifier 2: 1.3.6.1.2.1.1.4 (SNMPv2-MIB::sysContact) Value: NULL

36

Thao tác GetBulkRequest



- ❖ Một SNMP agent tại địa chỉ 192.168.1.1 có hỗ trợ toàn bộ cây MIB-2.

Cho biết các OID được trả về khi thực hiện lệnh SNMP như sau:

```
snmpbulkget -v 2c -c public -Cn2 -Cr3 192.168.1.1  
.1.3.6.1.2.1.4.2 .1.3.6.1.2.1.4.7 .1.3.6.1.2.1.4.20.1.1  
.1.3.6.1.2.1.4.22.1.1
```

-Thực hiện: Nhận được 7 OID như sau:

```
-Getnext .1.3.6.1.2.1.4.2->1.4.2.0  
-Getnext .1.3.6.1.2.1.4.7->1.4.7.0  
-Getnext .1.3.6.1.2.1.4.20.1.1 ->20.1.1.0  
-Getnext .1.3.6.1.2.1.4.20.1.1.0 -> 20.1.2.0  
-Getnext .1.3.6.1.2.1.4.20.1.2.0 -> 20.1.3.0  
-Getnext .1.3.6.1.2.1.4.22.1.1 -> 22.1.1.0  
-Getnext .1.3.6.1.2.1.4.22.1.1.0 -> 22.1.2.0  
-Getnext .1.3.6.1.2.1.4.22.1.2.0 -> 22.1.3.0
```

37

	tcpConnState	tcpConnLocalAddress	tcpConnLocalPort	tcpConnRemAddress	tcpConnRemPort
0	listen	0.0.0.0	135	0.0.0.0	43
246	listen	0.0.0.0	445	0.0.0.0	20
648	listen	0.0.0.0	1025	0.0.0.0	6
204	listen	0.0.0.0	1026	0.0.0.0	20
652	listen	192.168.1.1	139	0.0.0.0	8
324					

Cho bảng tcpConnTable như hình trên. Chỉ dùng các lệnh GetNext hãy truy xuất toàn bộ thông tin trong bảng này. Cho biết thứ tự các OID nhận được

Biết OID của tcpConnTable = 1.3.6.1.2.1.6.13

38



Getnext 6.13.1

6.13.1.1.0.0.0.0.135.0.0.0.0.43246-> listen

Getnext 6.13.1.1.0.0.0.0.135.0.0.0.0.43246

6.13.1.1.0.0.0.0.445.0.0.0.0.20648 -> Listen

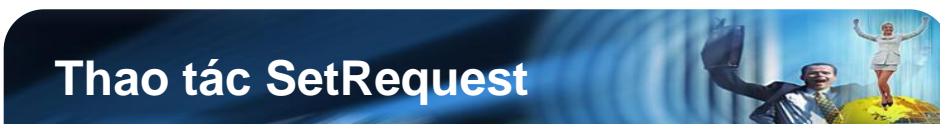
....

Getnext

.1.3.6.1.2.1.6.13.1.1.192.168.1.1.139.0.0.0.0.83
24

.1.3.6.1.2.1.6.13.1.2.0.0.0.0.135.0.0.0.0.43246 =
0.0.0.0

39



Thao tác SetRequest

❖ Chức năng: Giám giá trị cho một hoặc nhiều đối tượng MIB trên SNMP agent

■ Cấu trúc PDU:

<i>PDU type</i> 2	<i>Request ID</i> x	<i>ErrorStatus</i> 0	<i>ErrorIndex</i> 0	<i>OID 1</i> Value 1	<i>OID 2</i> Value 2	<i>OID n</i> Value n
----------------------	------------------------	-------------------------	------------------------	-------------------------	-------------------------	------	-------------------------

■ Chỉ thực hiện được thao tác set khi community string có quyền write

40

Thao tác SetRequest



■ Câu lệnh SNMP:

```
snmpset -v 2c -c private localhost system.sysContact.0 s "PTIT-HCM"
```

■ Kết quả bắt gói (bỏ IP header)

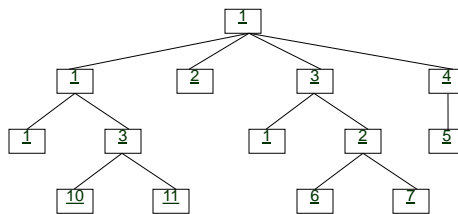
```
User Datagram Protocol:  
  Src Port: 34102 (34102)  
  Dst Port: snmp (161)  
  Length: 55  
Simple Network Management Protocol Version: 2C (1)  
  Community: private  
  PDU type: SET (3)  
  Request Id: 0x34df1dbe  
  Error Status: NO ERROR (0) Error Index: 0  
  Object identifier 1: 1.3.6.1.2.1.1.5.0 (SNMPv2-  
  MIB::sysName.0)  
  Value: STRING: PTIT-HCM
```

41

Bài tập



- ❖ Một cây MIB với các OID được cho như hình vẽ. Mỗi đối tượng trong cây MIB đều có chứa thông tin. Hãy liệt kê theo thứ tự OID của các đối tượng được trả về khi thực hiện liên tiếp các lệnh GetNext, bắt đầu từ đối tượng 1



Hình I.3.3

Getnext 1 -> 1.1.1.0

Getnext 1.1.1.0 -> 1.1.3.10.0

Getnext 1.1.3.10.0 -> 1.1.3.11.0

Getnext 1.1.3.11.0 -> 1.2.0

Getnext 1.2.0 -> 1.3.1.0

Getnext 1.3.1.0 -> 1.3.2.6.0

Getnext 1.3.2.6.0 -> 1.3.2.7.0

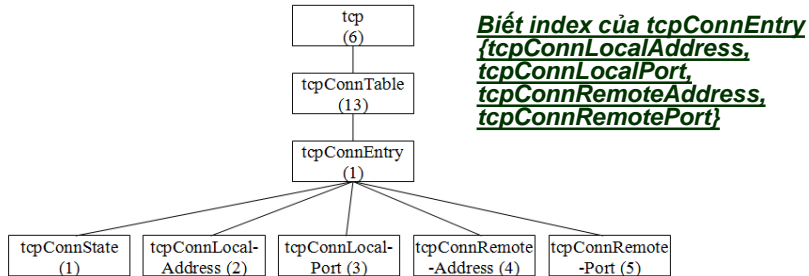
Getnext 1.3.2.7.0 -> 1.4.5.0

Getnext 1.4.5.0 -> Nosuchvalue

42

Bài tập

- ❖ Cho cây MIB của nhóm TCP như hình II.3.3a và bảng kết nối TCP như bảng II.3.3b. Hãy dùng các câu lệnh *GetNext* thích hợp để đọc tất cả các thông tin trong bảng này. Sắp xếp các thông tin theo thứ tự đọc được.



Hình II.3.3a

tcpConnState	tcpConnLocalAddress	tcpConnLocalPort	tcpConnRemoteAddress	tcpConnRemotePort
Closed	192.168.1.100	1876	66.171.20.10	25
SynReceived	203.162.44.162	80	120.18.42.6	1865
Established	192.168.2.1	1890	189.34.12.1	80
Established	192.168.1.12	2011	227.203.17.245	80

43

Câu hỏi

- ❖ Cấu trúc bản tin SNMP ứng với các PDU get, getnext và response. Giải thích ý nghĩa từng phần trong cấu trúc này.
- ❖ Sử dụng Getnext để truy xuất lần lượt các đối tượng trên cây MIB theo thứ tự. Chú ý các đối tượng dạng bảng, dùng các trường index để xác định giá trị từng thành phần trong bảng.

44



SNMPv2 và SNMPv3

45



SNMP v1

❖ RFC 1155 (1990)

- Structure and Identification of Management Information for TCP/IP-based Internets

❖ RFC 1157 (1990)

- Simple Network Management Protocol

❖ RFC 1213 (1991)

- Management Information Base v2 (MIB-2)
- Chỉ gồm 4 thao tác: get, getnext, set và trap
- Hiệu suất thấp, kém bảo mật

46

SNMPv2



❖ SNMP v2:

- Bổ sung hai thao tác: getbulk, notification, inform, report
- Bổ sung một số kiểu dữ liệu trong SMI (Integer32, Counter32, Gauge32 ...)
- Bổ sung cơ chế bảo mật

❖ SNMP 2c (community-based):

- Cơ chế hoạt động và kiểu dữ liệu tương tự SNMPv2, trừ mô hình bảo mật, vẫn dùng community string

47

SNMPv3



❖ RFC 3411, RFC 3412, RFC 3413, RFC 3414, RFC 3415.

❖ Mục tiêu: Bổ sung cơ chế bảo mật cho giao thức QLM với 3 trọng tâm:

- Xác thực user (user authentication)
- Xác thực thông tin (message authentication)
- Mã hóa thông tin (message encryption)

48



GIAO THỨC TELNET

49



Giao thức telnet

- ❖ Máy tính với giao diện consle
- ❖ Người ta sử dụng các hệ thống dựa trên nền văn bản (text-based)
- ❖ Thường là các đầu cuối hiển thị kết nối vào máy tính trung tâm
- ❖ Telnet cung cấp một phương pháp chuẩn để mô phỏng các thiết bị đầu cuối text-based thông qua mạng dữ liệu

50

Giao thức telnet



- ❖ Máy tính với giao diện consle
- ❖ Giao thức và phần mềm client sử dụng giao thức đều được gọi chung là Telnet.
- ❖ Kết nối sử dụng Telnet là phiên VTY (Virtual Terminal) hoặc kết nối VTY.

51

Giao thức telnet



- ❖ Máy tính với giao diện consle
- ❖ Trong Windows có thể chạy Telnet từ dấu nhắc lệnh.
- ❖ Các ứng dụng khác có thể làm việc như Telnet client: HyperTerminal, Minicom, TeraTerm.

52

Giao thức telnet



- ❖ Máy tính với giao diện console
- ❖ Telnet là một giao thức chủ/khách, quy định cách thiết lập và hủy bỏ một phiên VTY. Mỗi lệnh Telnet chứa ít nhất 2 byte:
 - ❖ Byte đầu chứa một ký tự đặc biệt gọi là ký tự IAC (Interpret as Command).
 - ❖ Byte còn lại là lệnh.

53

Giao thức telnet



- ❖ Một số lệnh Telnet:
 - ❖ AYT (Are You There) – cho phép người dùng yêu cầu một số thông tin xuất hiện trên màn hình terminal để xác định phiên VTY đang được kích hoạt.
 - ❖ EL (Erase Line) – xóa toàn bộ text từ dòng hiện thời.

54

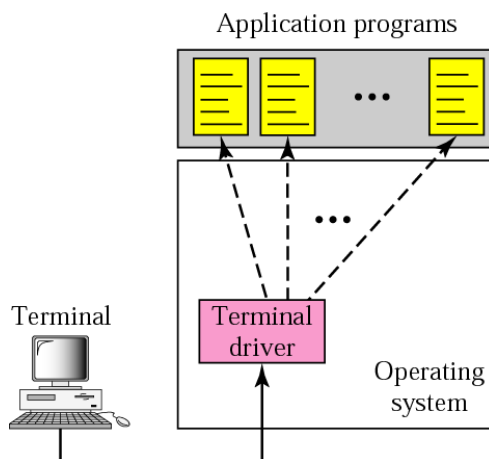
Giao thức telnet - Bảo mật



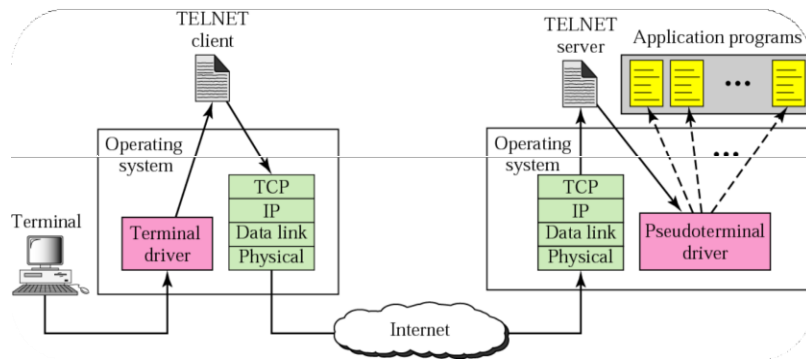
- ❖ Hỗ trợ xác thực người dùng
- ❖ Không hỗ trợ truyền dữ liệu được mã hóa
 - ❖ Dữ liệu trao đổi trong các phiên Telnet được truyền dưới dạng văn bản (plaintext) qua mạng
 - ❖ Giao thức SSH (Secure Shell) cung cấp giải pháp an toàn hơn cho truy cập máy chủ
 - ❖ SSH hỗ trợ xác thực mạnh hơn Telnet, hỗ trợ truyền dữ liệu được mã hóa.

55

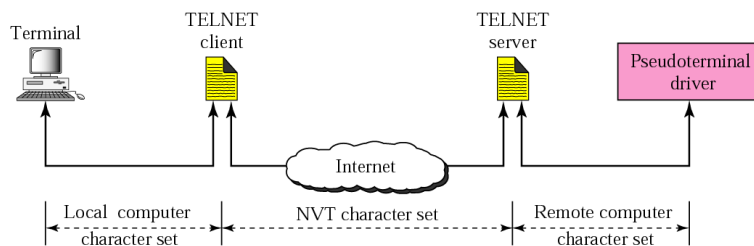
Mô hình sử dụng TELNET



Telnet client-server



Tập lệnh cho Network VT





- Data được truyền bằng bảng mã với bit cao là 0

0							
---	--	--	--	--	--	--	--



- Các ký tự control được truyền bằng bảng mã với bit cao là 1

1							
---	--	--	--	--	--	--	--



Các ký tự control thông dụng

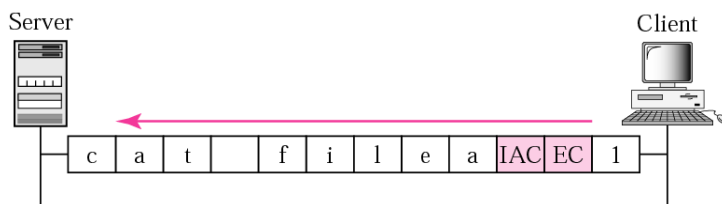


Character	Decimal	Binary	Meaning
EOF	236	11101100	End of file
EOR	239	11101111	End of record
SE	240	11110000	Suboption end
NOP	241	11110001	No operation
DM	242	11110010	Data mark
BRK	243	11110011	Break
IP	244	11110100	Interrupt process
AO	245	11110101	Abort output
AYT	246	11110110	Are you there?
EC	247	11110111	Erase character
EL	248	11111000	Erase line
GA	249	11111001	Go ahead
SB	250	11111010	Suboption begin
WILL	251	11111011	Agreement to enable option
WONT	252	11111100	Refusal to enable option
DO	253	11111101	Approval to option request
DONT	254	11111110	Denial of option request
IAC	255	11111111	Interpret (the next character) as control

Nhúng ký tự control



- Telnet dùng chung một kênh để trao đổi dữ liệu và điều khiển, các ký tự control được nhúng theo dòng dữ liệu
- Ví dụ như sau:



Tham số cho telnet



- Giữa Telnet client và server thỏa thuận với nhau các option trước khi tiến hành giao tiếp.

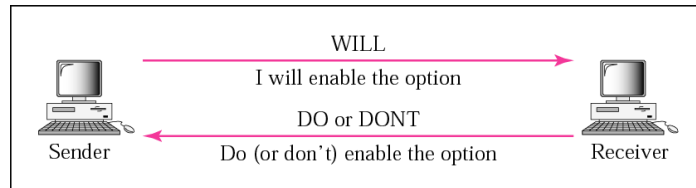
<i>Code</i>	<i>Option</i>	<i>Meaning</i>
0	Binary	Interpret as 8-bit binary transmission
1	Echo	Echo the data received on one side to the other
3	Suppress go ahead	Suppress go-ahead signals after data
5	Status	Request the status of TELNET
6	Timing mark	Define the timing marks
24	Terminal type	Set the terminal type
32	Terminal speed	Set the terminal speed
34	Line mode	Change to line mode

Các ký tự thỏa thuận các option



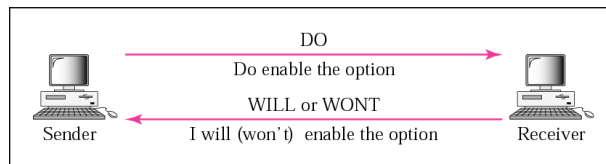
<i>Character</i>	<i>Decimal</i>	<i>Binary</i>	<i>Meaning</i>
WILL	251	11111011	1. Offering to enable 2. Accepting a request to enable
WONT	252	11111100	1. Rejecting a request to enable 2. Offering to disable 3. Accepting a request to disable
DO	253	11111101	1. Approving an offer to enable 2. Requesting to enable
DONT	254	11111110	1. Disapproving an offer to enable 2. Approving an offer to disable 3. Requesting to disable

Yêu cầu bật option (bị động)



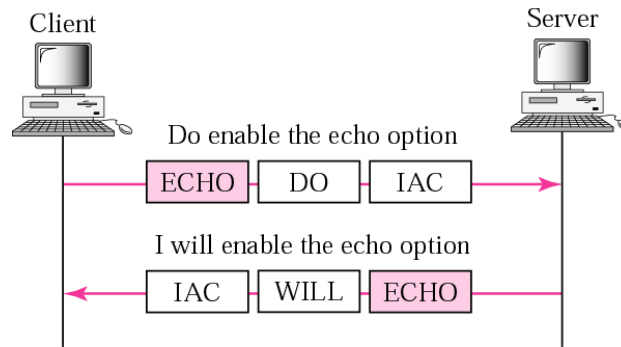
2

Yêu cầu bật option (chủ động)



66

Ví dụ bật option ECHO



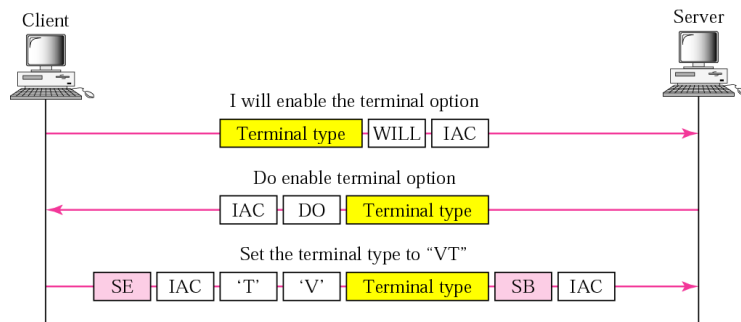
67

- Một số option đòi hỏi các tham số con để hoạt động. Các tham số con được giới hạn bởi ký tự SB và SE

<i>Character</i>	<i>Decimal</i>	<i>Binary</i>	<i>Meaning</i>
SE	240	11110000	Suboption end
SB	250	11111010	Suboption begin

68

Ví dụ option con



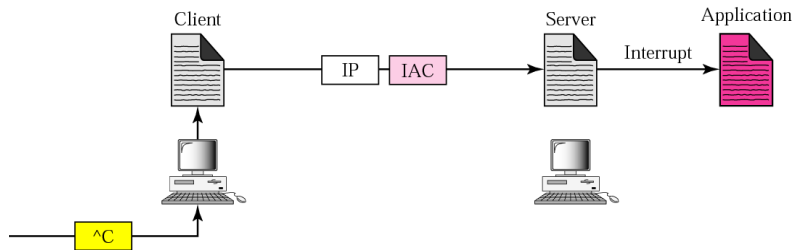
69

Ký tự điều khiển phía server

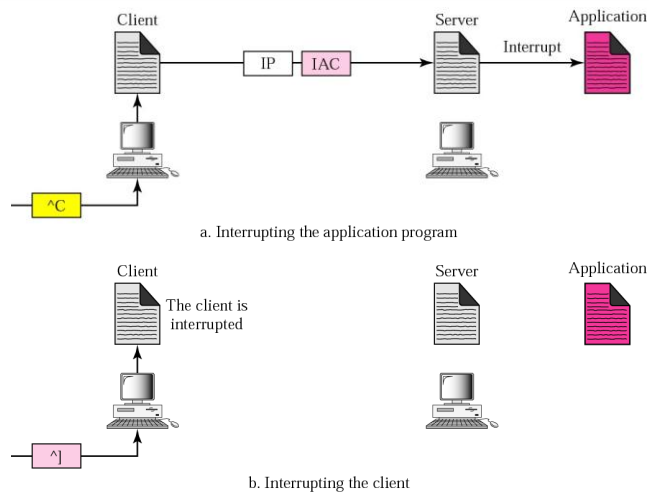
Character	Decimal	Binary	Meaning
IP	244	11110100	Interrupt process
AO	245	11110101	Abort output
AYT	246	11110110	Are you there?
EC	247	11110111	Erase the last character
EL	248	11111000	Erase line

70

VD ngắt ứng dụng đang chạy phía server



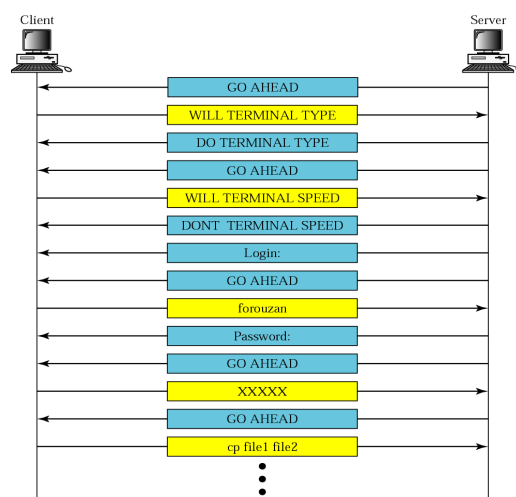
Ngắt ở phía client



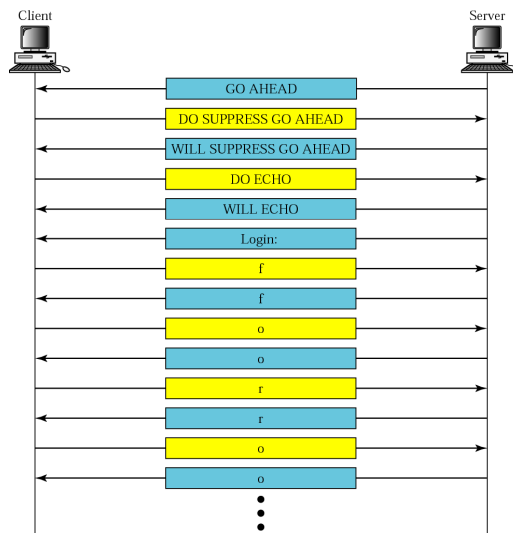
Chế độ hoạt động

- Telnet hoạt động ở 3 chế độ khác nhau:
 - Chế độ Default
 - Chế độ Character
 - Chế độ Line

Chế độ Default



Chế độ Character



75

TELNET – Nhược điểm

- ❑ TELNET là giao thức không mã hóa
- ❑ Hầu hết các ứng dụng sử dụng Telnet đều không dùng chứng thực
- ❑ Không an toàn khi kết nối thông qua đường mạng.



SSH (SECURE SHELL)



Hạn chế của các giao thức trước

- Các giao thức như Telnet, FTP, r-*,... thiếu khả năng trao đổi dữ liệu an toàn
- Dữ liệu hoặc lệnh được truyền theo định dạng không mã hóa trên đường truyền.
- Telnet: Username và password(nếu có) được truyền theo dạng văn bản thuần.

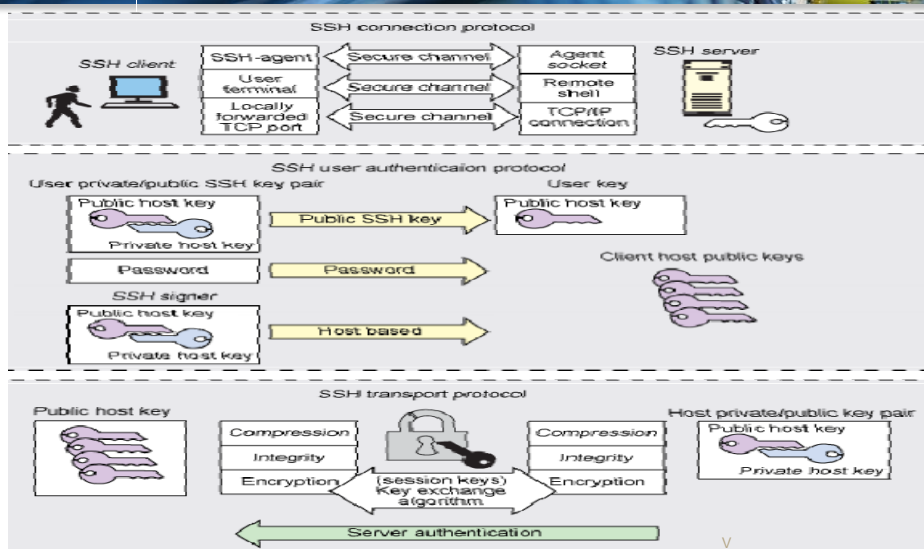


SSH - Tính năng

- ❑ Chứng thực: Chứng thực user và server.
- ❑ Mã hóa và nén dữ liệu.
- ❑ Kiểm tra tính toàn vẹn của dữ liệu(MD5, SHA-1)
- ❑ Hoạt động trên nền TCP

79

Các thành phần SSH



80

Các ứng dụng thường dùng SSH



- Copy file an toàn qua mạng: scp
- File server an toàn: SFTP
- Thực thi lệnh quản trị qua mạng: SSH
- ...

81

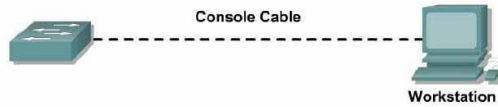
SSH & Key



- Khi client lần đầu kết nối SSH đến server, client sẽ ghi nhớ public key.
- Các kết nối lần sau nếu public key thay đổi thì có thể kết nối đang bị giả mạo (Man in the Middle)
- SSH client thông báo nếu có thay đổi public key

82

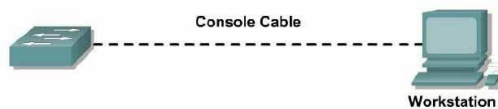
Telnet trên Cisco Switch



- Switch>enable
- Switch#show running-config
- **Thiết lập các thông số cho switch như hostname, enable password, console password và virtual terminal password.**

83

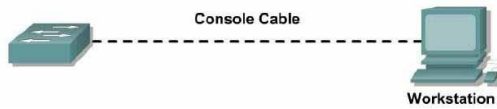
Telnet trên Cisco Switch



- *Switch>enable*
Switch#config terminal
Switch(config)#hostname Ou
Ou(config)#enable password cisco
Ou(config)#enable secret class
Ou(config)#line console 0
Ou(config-line)#password console
Ou(config-line)#login
Ou(config-line)#^Z

84

Telnet trên Cisco Switch



- *Switch>enable*
- **Cấu hình password cho các line vty**

OU#config terminal

OU(config)#line vty 0 4

OU(config-line)#password cisco

OU(config-line)#login

85

Q&A



86