Q1 SNMP (Section 9.9)

1 Point

Q1.1 SNMP community string (Section 9.9, Exercise 2)

0.5 Points

What is the difference in the output when running `snmpwalk` on the "server" host with "public" as the community string, versus "secret" as the community string, **and why**? Show output in each case, and explain.

Running snmpwalk on the server host with "secret" as the commun ity string will output much more information than using "public" as t he community string. Since it made a small amount of managemen t information available to the public but much more information av ailable to the designated host that we want to be able to monitor t he network condition.

server_public.png Download

```
ty2069@server:~$ snmpwalk -v 2c -c public router-int
iso.3.6.1.2.1.1.1.0 = STRING: "Linux router-int.lab9-ty2069.ch-geni-net.instagen
i.ucsd.edu 4.15.0-121-generic #123-Ubuntu SMP Mon Oct 5 16:16:40 UTC 2020 x86_64
"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.10
iso.3.6.1.2.1.1.3.0 = Timeticks: (1028) 0:00:10.28
iso.3.6.1.2.1.1.4.0 = STRING: "Me <me@example.org>"
iso.3.6.1.2.1.1.5.0 = STRING: "router-int.lab9-ty2069.ch-geni-net.instageni.ucsd
.edu"
iso.3.6.1.2.1.1.6.0 = STRING: "Sitting on the Dock of the Bay"
iso.3.6.1.2.1.1.7.0 = INTEGER: 72
iso.3.6.1.2.1.1.8.0 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.9 = OID: iso.3.6.1.6.3.13.3.1.3
iso.3.6.1.2.1.1.9.1.2.10 = OID: iso.3.6.1.2.1.92
iso.3.6.1.2.1.1.9.1.3.1 = STRING: "The MIB for Message Processing and Dispatchin
g."
iso.3.6.1.2.1.1.9.1.3.2 = STRING: "The management information definitions for th
e SNMP User-based Security Model."
iso.3.6.1.2.1.1.9.1.3.3 = STRING: "The SNMP Management Architecture MIB."
iso.3.6.1.2.1.1.9.1.3.4 = STRING: "The MIB module for SNMPv2 entities"
iso.3.6.1.2.1.1.9.1.3.5 = STRING: "View-based Access Control Model for SNMP."
iso.3.6.1.2.1.1.9.1.3.6 = STRING: "The MIB module for managing TCP implementatio
ns"
iso.3.6.1.2.1.1.9.1.3.7 = STRING: "The MIB module for managing IP and ICMP imple
mentations"
iso.3.6.1.2.1.1.9.1.3.8 = STRING: "The MIB module for managing UDP implementatio
ns"
iso.3.6.1.2.1.1.9.1.3.9 = STRING: "The MIB modules for managing SNMP Notificatio
n, plus filtering."
iso.3.6.1.2.1.1.9.1.3.10 = STRING: "The MIB module for logging SNMP Notification
s."
iso.3.6.1.2.1.1.9.1.4.1 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.2 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.3 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.4 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.5 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.6 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.7 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.8 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.9 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.10 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.25.1.1.0 = Timeticks: (16948729) 1 day, 23:04:47.29
iso.3.6.1.2.1.25.1.2.0 = Hex-STRING: 07 E4 0C 04 14 1A 17 00 2D 05 00
iso.3.6.1.2.1.25.1.3.0 = INTEGER: 393216
iso.3.6.1.2.1.25.1.4.0 = STRING: "root=UUID=5fac4c69-51f6-47af-9773-a0d722426942
 ro console=ttyS0,115200 root=/dev/xvda1 ro console=hvc0 xencons=tty apparmor=0
se"
iso.3.6.1.2.1.25.1.5.0 = Gauge32: 1
iso.3.6.1.2.1.25.1.6.0 = Gauge32: 86
iso.3.6.1.2.1.25.1.7.0 = INTEGER: 0
iso.3.6.1.2.1.25.1.7.0 = No more variables left in this MIB View (It is past the
 end of the MIB tree)
```

server_secret.png Download

```
ty2069@server:~$ snmpwalk -v 2c -c secret router-int
iso.3.6.1.2.1.1.1.0 = STRING: "Linux router-int.lab9-ty2069.ch-geni-net.instagen
i.ucsd.edu 4.15.0-121-generic #123-Ubuntu SMP Mon Oct 5 16:16:40 UTC 2020 x86_64
"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.10
iso.3.6.1.2.1.1.3.0 = Timeticks: (1061) 0:00:10.61
iso.3.6.1.2.1.1.4.0 = STRING: "Me <me@example.org>"
iso.3.6.1.2.1.1.5.0 = STRING: "router-int.lab9-ty2069.ch-geni-net.instageni.ucsd
.edu"
iso.3.6.1.2.1.1.6.0 = STRING: "Sitting on the Dock of the Bay"
iso.3.6.1.2.1.1.7.0 = INTEGER: 72
iso.3.6.1.2.1.1.8.0 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.9 = OID: iso.3.6.1.6.3.13.3.1.3
iso.3.6.1.2.1.1.9.1.2.10 = OID: iso.3.6.1.2.1.92
iso.3.6.1.2.1.1.9.1.3.1 = STRING: "The MIB for Message Processing and Dispatchin
g."
iso.3.6.1.2.1.1.9.1.3.2 = STRING: "The management information definitions for th
e SNMP User-based Security Model."
iso.3.6.1.2.1.1.9.1.3.3 = STRING: "The SNMP Management Architecture MIB."
iso.3.6.1.2.1.1.9.1.3.4 = STRING: "The MIB module for SNMPv2 entities"
iso.3.6.1.2.1.1.9.1.3.5 = STRING: "View-based Access Control Model for SNMP."
iso.3.6.1.2.1.1.9.1.3.6 = STRING: "The MIB module for managing TCP implementatio
ns"
iso.3.6.1.2.1.1.9.1.3.7 = STRING: "The MIB module for managing IP and ICMP imple
mentations"
iso.3.6.1.2.1.1.9.1.3.8 = STRING: "The MIB module for managing UDP implementatio
ns"
iso.3.6.1.2.1.1.9.1.3.9 = STRING: "The MIB modules for managing SNMP Notificatio
n, plus filtering."
iso.3.6.1.2.1.1.9.1.3.10 = STRING: "The MIB module for logging SNMP Notification
s."
iso.3.6.1.2.1.1.9.1.4.1 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.2 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.3 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.4 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.5 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.6 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.7 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.8 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.9 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.10 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.2.1.0 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.1.1 = INTEGER: 1
iso.3.6.1.2.1.2.2.1.1.2 = INTEGER: 2
iso.3.6.1.2.1.2.2.1.1.3 = INTEGER: 3
iso.3.6.1.2.1.2.2.1.1.4 = INTEGER: 4
iso.3.6.1.2.1.2.2.1.1.5 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.2.1 = STRING: "lo"
iso.3.6.1.2.1.2.2.1.2.2 = STRING: "eth0"
iso.3.6.1.2.1.2.2.1.2.3 = STRING: "eth1"
iso.3.6.1.2.1.2.2.1.2.4 = STRING: "eth2"
iso.3.6.1.2.1.2.2.1.2.5 = STRING: "eth3"
iso.3.6.1.2.1.2.2.1.3.1 = INTEGER: 24
iso.3.6.1.2.1.2.2.1.3.2 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.3 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.4 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.5 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.4.1 = INTEGER: 65536
iso.3.6.1.2.1.2.2.1.4.2 = INTEGER: 1500
```

What is the difference in the output when running `snmpwalk` with "secret" as the community string on the "server" host, versus with "secret" as the community string on the "romeo" host, **and why**? Show output in each case, and explain.

On romeo host, we will have No Response as output since in the r outer's snmpd configuration file we set up the access control rule t hat only allows the server host could access the secret informatio n.

▾ romeo_secret.png Download

```
ty2069@romeo:~$ snmpwalk -v 2c -c secret router-int
Timeout: No Response from router-int
```

▾ server_secret.png Download

```
ty2069@server:~$ snmpwalk -v 2c -c secret router-int
iso.3.6.1.2.1.1.1.0 = STRING: "Linux router-int.lab9-ty2069.ch-geni-net.instagen
i.ucsd.edu 4.15.0-121-generic #123-Ubuntu SMP Mon Oct 5 16:16:40 UTC 2020 x86_64
"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.10
iso.3.6.1.2.1.1.3.0 = Timeticks: (1061) 0:00:10.61
iso.3.6.1.2.1.1.4.0 = STRING: "Me <me@example.org>"
iso.3.6.1.2.1.1.5.0 = STRING: "router-int.lab9-ty2069.ch-geni-net.instageni.ucsd
.edu"
iso.3.6.1.2.1.1.6.0 = STRING: "Sitting on the Dock of the Bay"
iso.3.6.1.2.1.1.7.0 = INTEGER: 72
iso.3.6.1.2.1.1.8.0 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.9 = OID: iso.3.6.1.6.3.13.3.1.3
iso.3.6.1.2.1.1.9.1.2.10 = OID: iso.3.6.1.2.1.92
iso.3.6.1.2.1.1.9.1.3.1 = STRING: "The MIB for Message Processing and Dispatchin
g."
iso.3.6.1.2.1.1.9.1.3.2 = STRING: "The management information definitions for th
e SNMP User-based Security Model."
iso.3.6.1.2.1.1.9.1.3.3 = STRING: "The SNMP Management Architecture MIB."
iso.3.6.1.2.1.1.9.1.3.4 = STRING: "The MIB module for SNMPv2 entities"
iso.3.6.1.2.1.1.9.1.3.5 = STRING: "View-based Access Control Model for SNMP."
iso.3.6.1.2.1.1.9.1.3.6 = STRING: "The MIB module for managing TCP implementatio
ns"
iso.3.6.1.2.1.1.9.1.3.7 = STRING: "The MIB module for managing IP and ICMP imple
mentations"
iso.3.6.1.2.1.1.9.1.3.8 = STRING: "The MIB module for managing UDP implementatio
ns"
iso.3.6.1.2.1.1.9.1.3.9 = STRING: "The MIB modules for managing SNMP Notificatio
n, plus filtering."
iso.3.6.1.2.1.1.9.1.3.10 = STRING: "The MIB module for logging SNMP Notification
s."
iso.3.6.1.2.1.1.9.1.4.1 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.2 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.3 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.4 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.5 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.6 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.7 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.8 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.9 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.10 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.2.1.0 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.1.1 = INTEGER: 1
iso.3.6.1.2.1.2.2.1.1.2 = INTEGER: 2
iso.3.6.1.2.1.2.2.1.1.3 = INTEGER: 3
iso.3.6.1.2.1.2.2.1.1.4 = INTEGER: 4
iso.3.6.1.2.1.2.2.1.1.5 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.2.1 = STRING: "lo"
iso.3.6.1.2.1.2.2.1.2.2 = STRING: "eth0"
iso.3.6.1.2.1.2.2.1.2.3 = STRING: "eth1"
iso.3.6.1.2.1.2.2.1.2.4 = STRING: "eth2"
iso.3.6.1.2.1.2.2.1.2.5 = STRING: "eth3"
iso.3.6.1.2.1.2.2.1.3.1 = INTEGER: 24
iso.3.6.1.2.1.2.2.1.3.2 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.3 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.4 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.3.5 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.4.1 = INTEGER: 65536
iso.3.6.1.2.1.2.2.1.4.2 = INTEGER: 1500
```

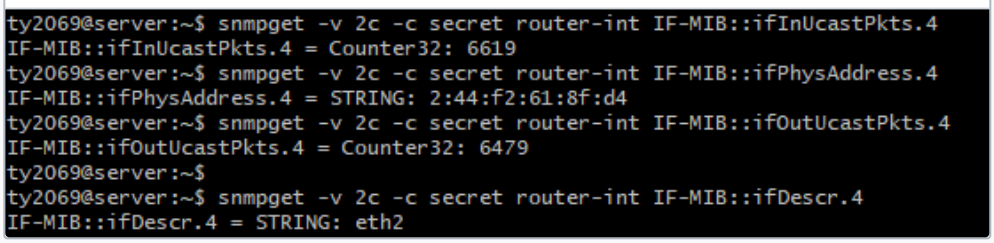
Q1.2 Retrieving SNMP data (Section 9.9, Exercise 1)

0.5 Points

Show the output of the `snmpget` commands.

▾ snmpget.png

▾ snmpget.png Download



Explain what each of the values you retrieved using `snmpget` means: `ifDescr`, `ifInUcastPkts`, `ifPhysAddress`, `ifOutUcastPkts`.

(You can quote directly from the contents of `/usr/share/snmp/mibs/ietf/IF-MIB`.)

ifDescr explanation:

"A textual string containing information about the
interface. This string should include the name of the
manufacturer, the product name and the version of the
interface hardware/software.

ifInUcastPkts explanation:

"The number of packets, delivered by this sub-layer to a
higher (sub-)layer, which were not addressed to a multicast
or broadcast address at this sub-layer.

Discontinuities in the value of this counter can occur at
re-initialization of the management system, and at other
times as indicated by the value of
ifCounterDiscontinuityTime."

ifPhysAddress explanation:

The interface's address at its protocol sub-layer. For
example, for an 802.x interface, this object normally
contains a MAC address. The interface's media-specific MI
B
must define the bit and byte ordering and the format of the
value of this object. For interfaces which do not have such

an address (e.g., a serial line), this object should contain an octet string of zero length.

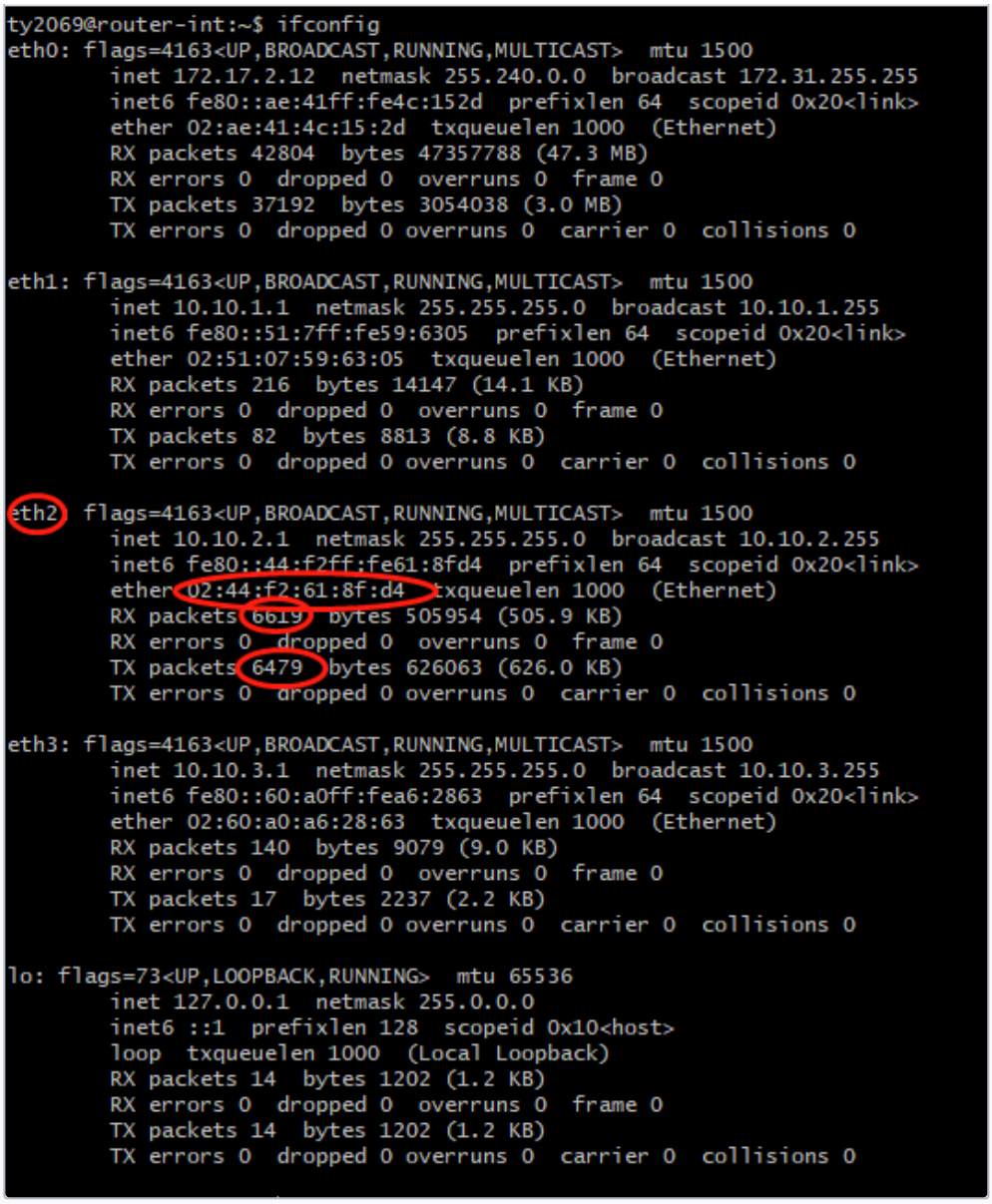
ifOutUcastPkts explanation:

The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

Show the relevant section of the `ifconfig` output on the router. Annotate your screenshot: circle the interface information you collected using SNMP.

router-ifconfig.png Download



Q2 Telnet and SSH (Section 9.10, Exercise 4)

2 Points

Q2.1 telnet

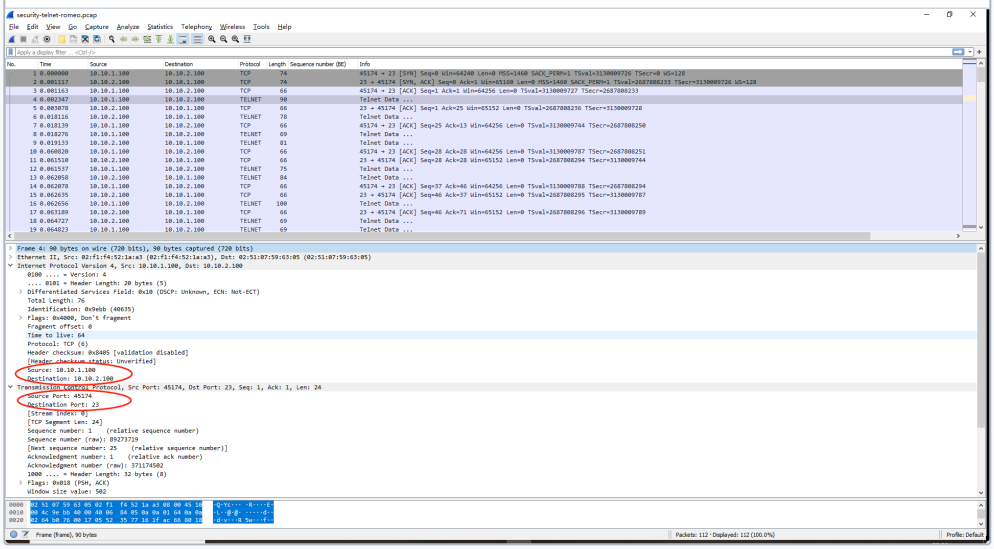
1 Point

In the packet capture of the `telnet` experiment, which of the following can you read from the captured packets (i.e. not encrypted)? Select all that apply:

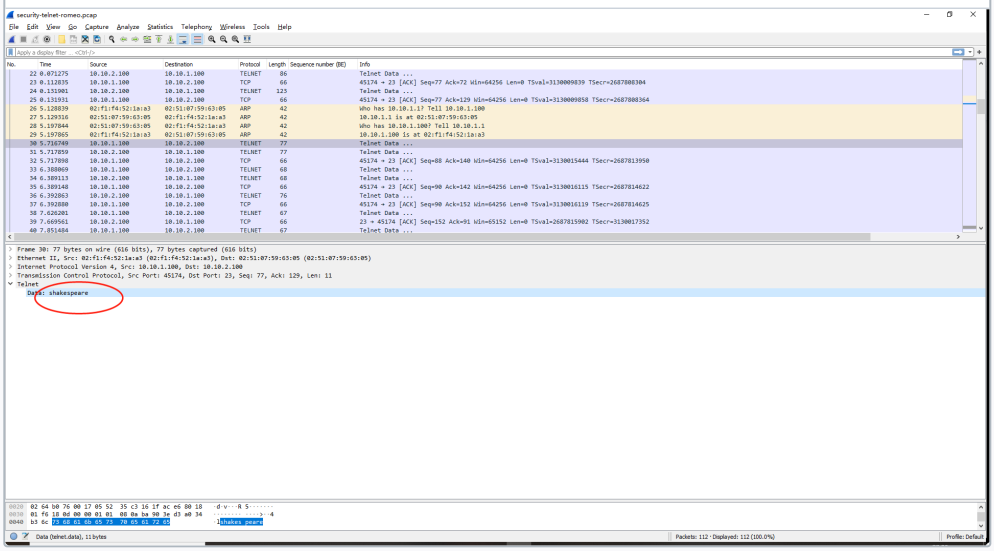
- [x] username and password
- [x] IP headers (e.g. source and destination IP address)
- [x] TCP headers (e.g. source and destination port)
- [x] session data (e.g. commands that the user runs, like `date`, and command output)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

▾ telnet1.png Download

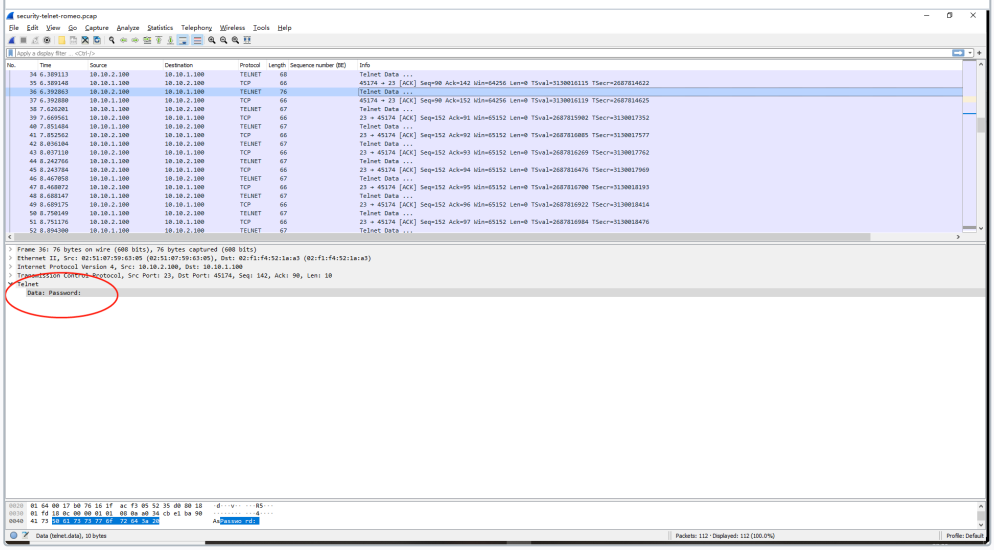


▾ telnet2.png Download



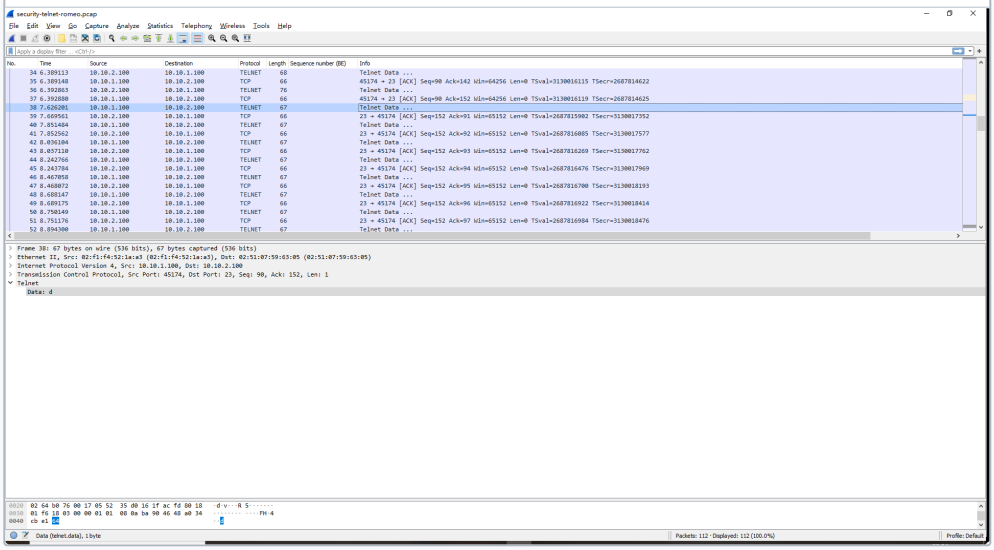
telnet3.png

Download



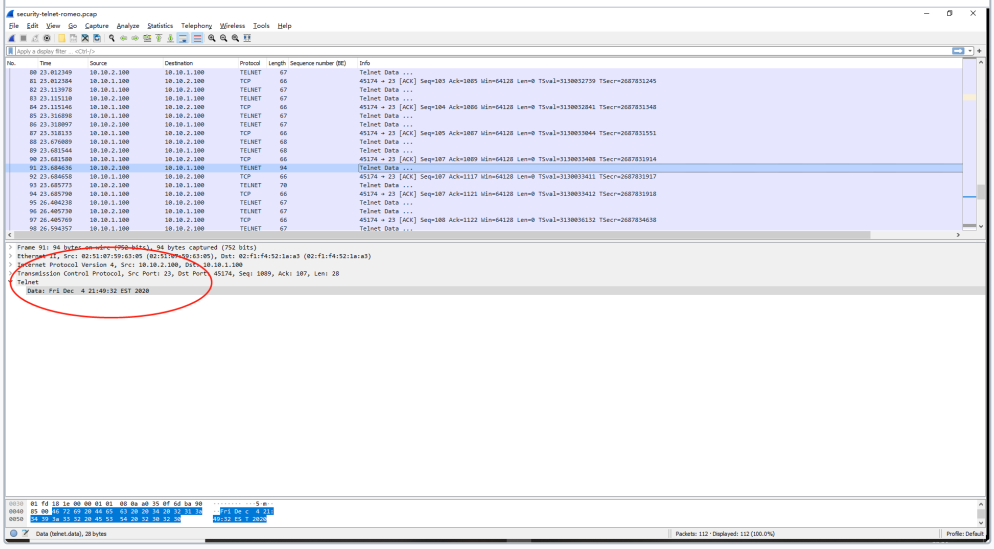
telnet4.png

Download



telnet5.png

Download



Q2.2 SSH

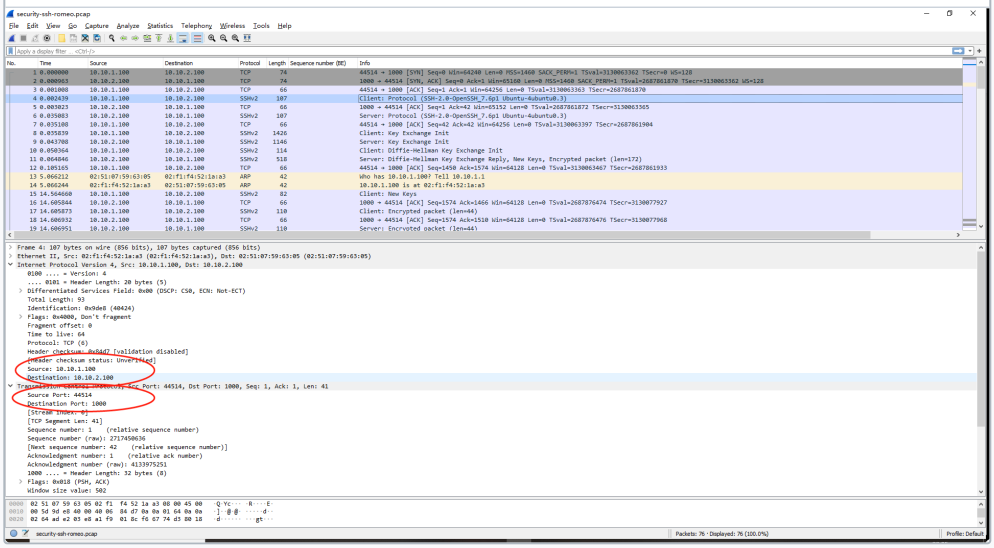
1 Point

In the packet capture of the SSH experiment, which of the following can you read from the captured packets (i.e. not encrypted)? Select all that apply:

- [ ] username and password
- [x] IP headers (e.g. source and destination IP address)
- [x] TCP headers (e.g. source and destination port)
- [ ] session data (e.g. commands that the user runs, like `date`, and command output)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

ssh.png Download



Q3 FTP and SFTP (Section 9.10, Exercise 6)

2 Points

Q3.1 FTP

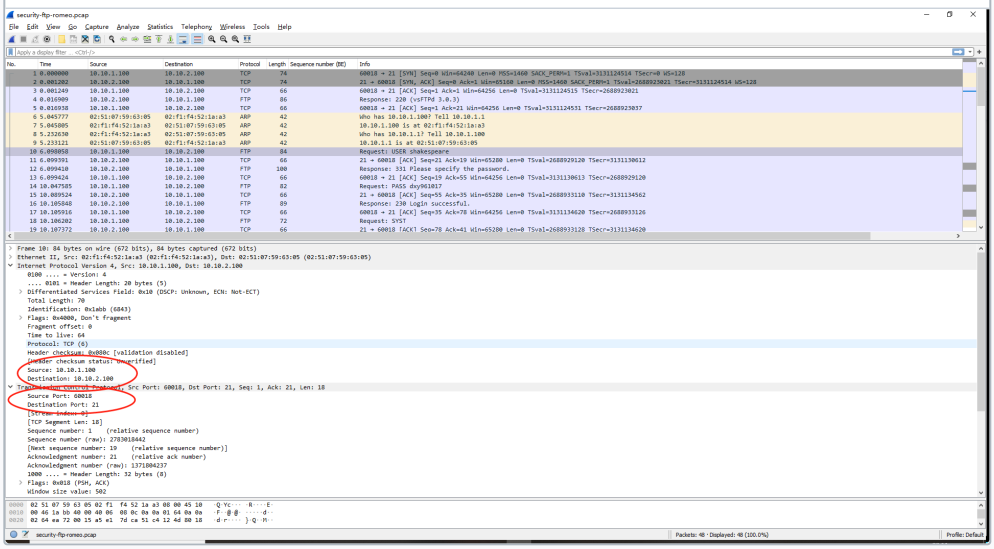
1 Point

In the packet capture of the FTP experiment, which of the following can you read from the captured packets (i.e. not encrypted)? Select all that apply:

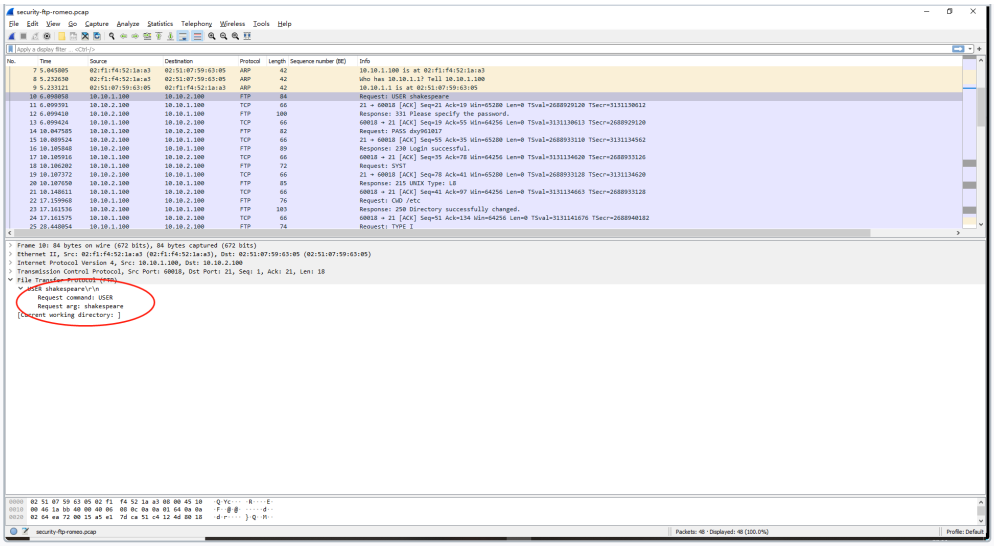
- [x] username and password
- [x] IP headers (e.g. source and destination IP address)
- [x] TCP headers (e.g. source and destination port)
- [x] session data (e.g. the name of the file that the user retrieves, the file contents)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

▾ ftp1.png Download

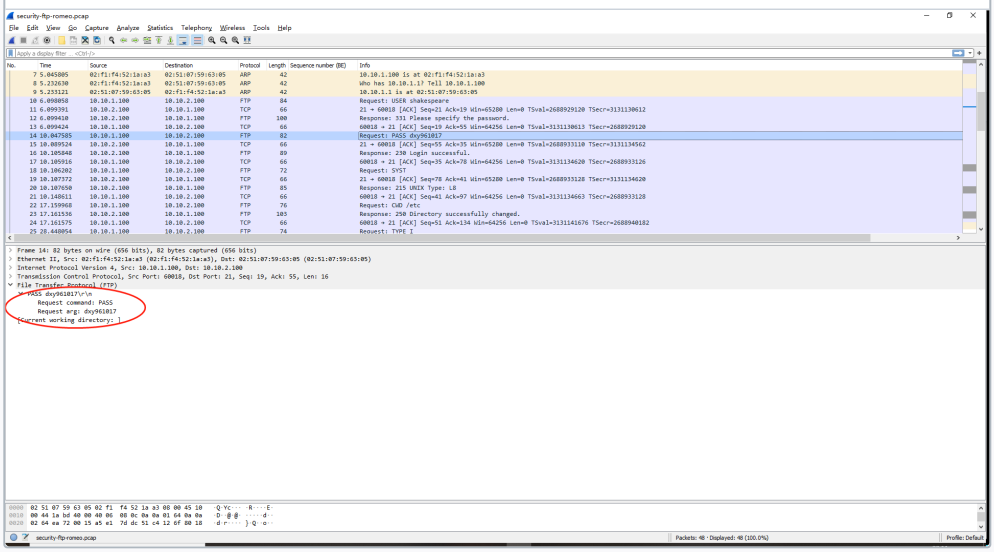


▾ ftp2.png Download



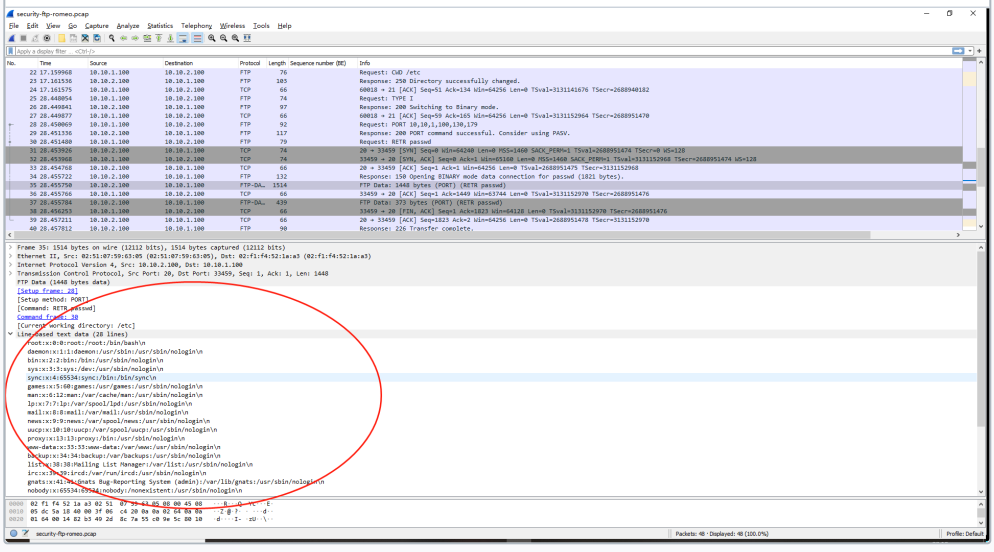
ftp3.png

Download



ftp4.png

Download



Q3.2 SFTP

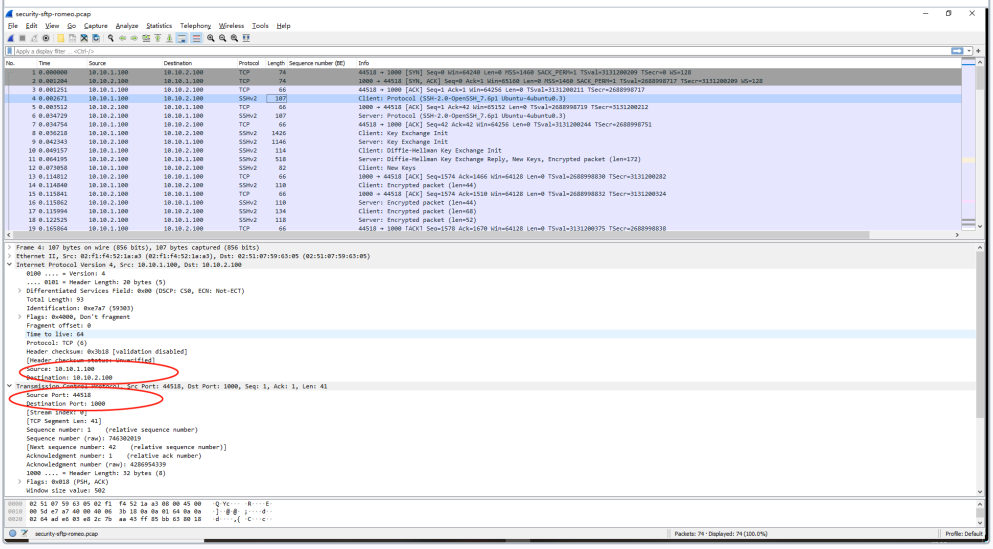
1 Point

In the packet capture of the SFTP experiment, which of the following can you read from the captured packets (i.e. not encrypted)? Select all that apply:

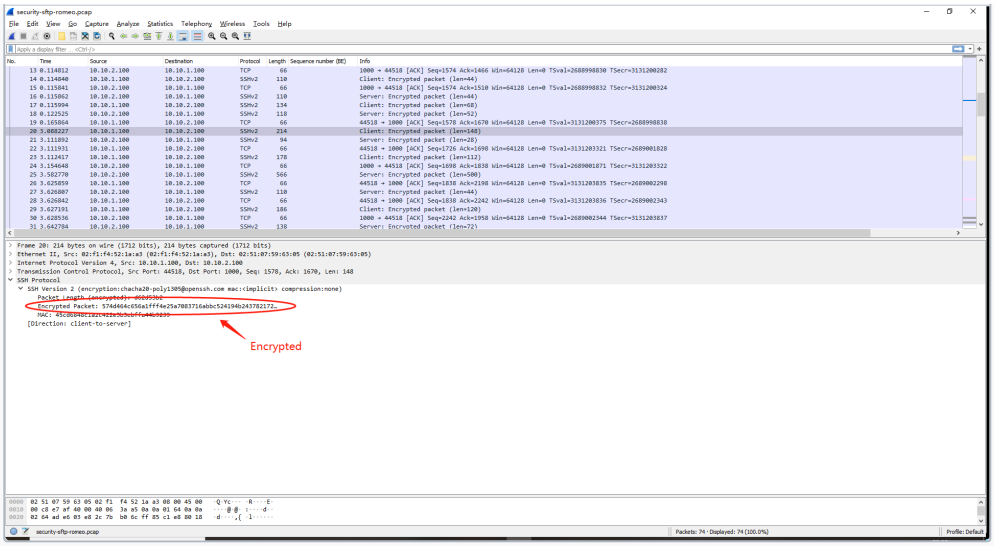
- [ ] username and password
- [x] IP headers (e.g. source and destination IP address)
- [x] TCP headers (e.g. source and destination port)
- [ ] session data (e.g. the name of the file that the user retrieves, the file contents)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

▾ sftp1.png Download



▾ sftp2.png Download



Q4 HTTP and HTTPS

2 Points

Q4.1 HTTP

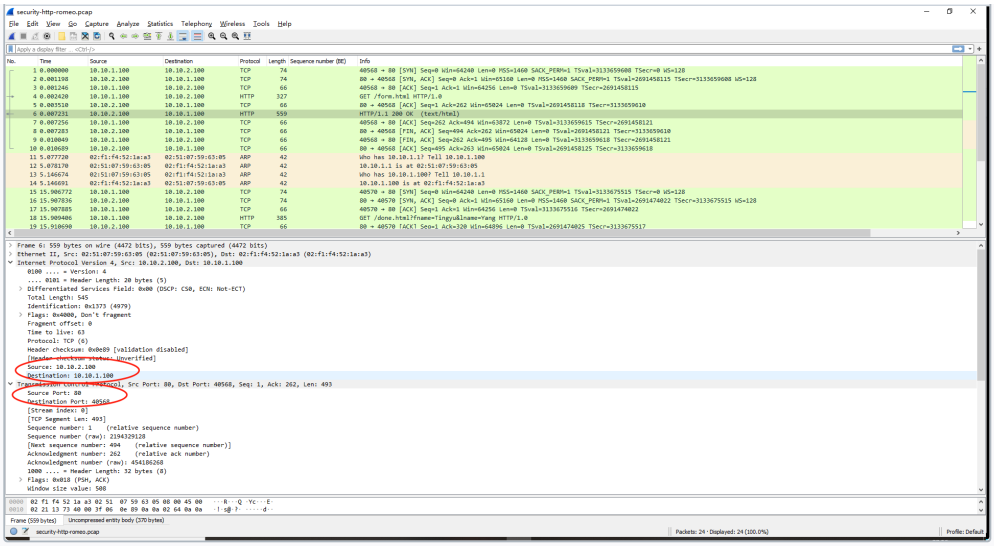
1 Point

In the packet capture of the HTTP experiment, which of the following can you read from the captured packets (i.e. not encrypted)? Select all that apply:

- [x] IP headers (e.g. source and destination IP address)
- [x] TCP headers (e.g. source and destination port)
- [x] contents of the HTTP GET (e.g. the name of the page you visited, `form.html`)
- [x] session data (e.g. the data you entered into the form)

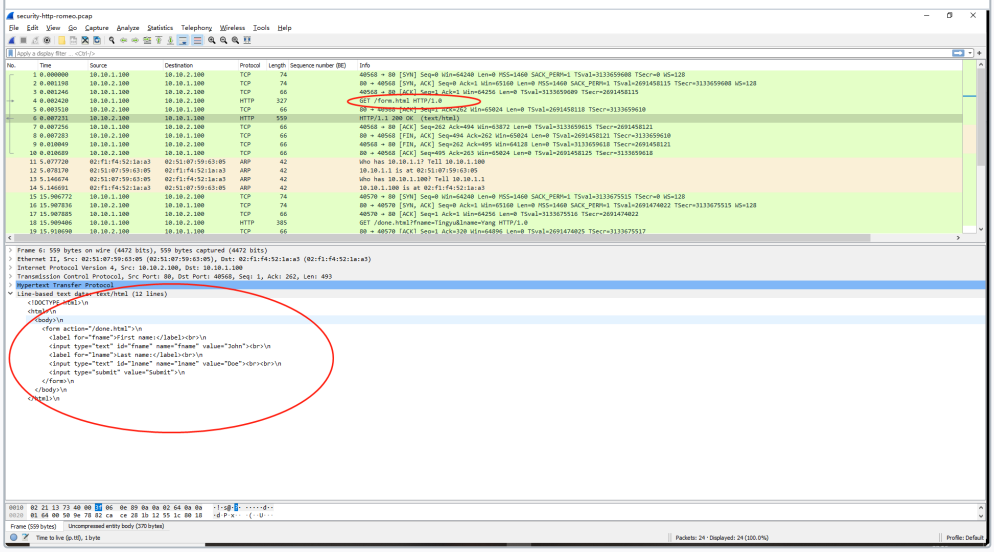
To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

▾ http1.png Download



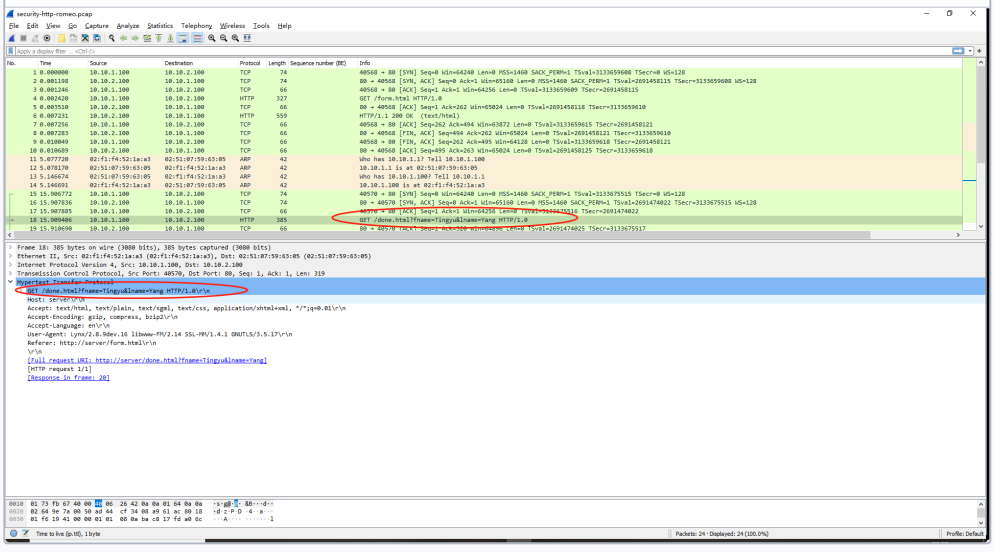
http2.png

Download



http3.png

Download



Q4.2 HTTPS

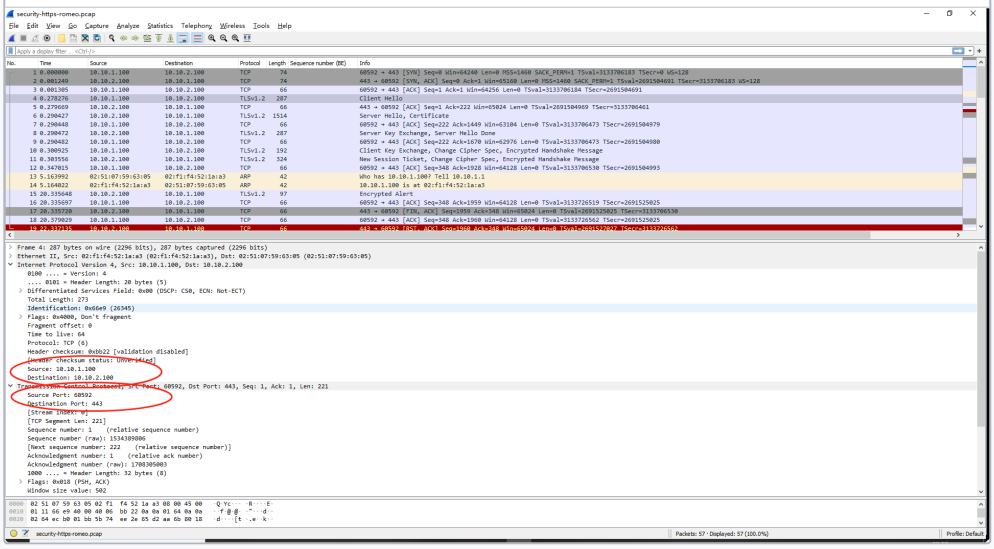
1 Point

In the packet capture of the HTTPS experiment, which of the following can you read from the captured packets (i.e. not encrypted)? Select all that apply:

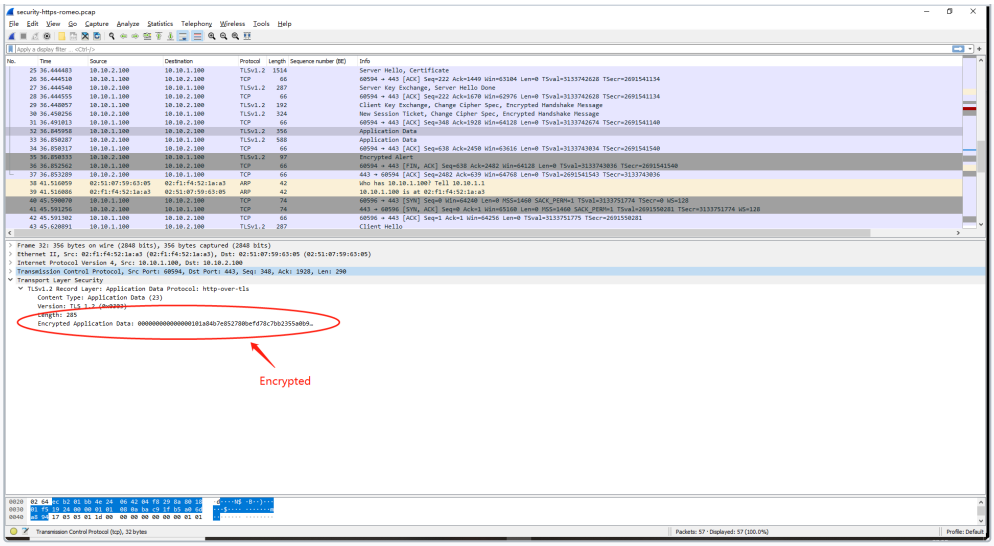
- [x] IP headers (e.g. source and destination IP address)
- [x] TCP headers (e.g. source and destination port)
- [ ] contents of the HTTP GET (e.g. the name of the page you visited, `form.html`)
- [ ] session data (e.g. the data you entered into the form)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

▾ https1.png Download



▾ https2.png Download



Q5 Network layer security

2 Points

Q5.1 VPN traffic at external router

1 Point

In the packet capture on the *external router*, which of the following can you read from the captured packets?

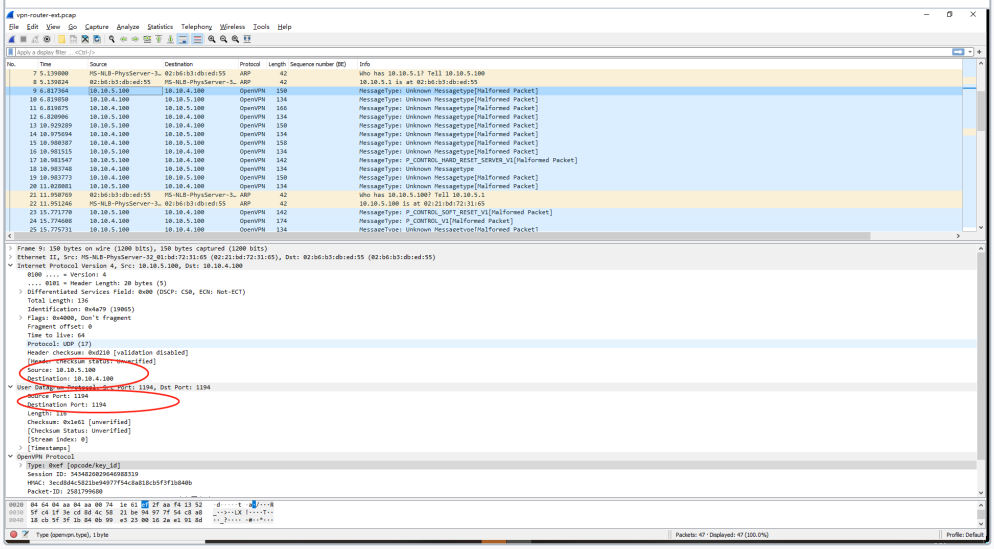
This information would be visible to a potential eavesdropper located somewhere along the network path **between the client and the VPN server**.

Select all that apply:

- [x] IP address of "juliet" on the external network (not the address on the VPN tunnel!)
- [x] IP address of the VPN server, "vpn" (i.e. the fact that the client is using this particular VPN server)
- [ ] IP address of the "server" node (i.e. the fact that the client is connecting to this particular FTP server)
- [x] UDP port 1194 (i.e. this connection uses the well-known port number of OpenVPN, so eavesdroppers can identify it as VPN traffic)
- [ ] TCP port 21 (i.e. this connection uses the well-known port number of FTP, so eavesdroppers can identify it as FTP traffic)
- [ ] Session data (e.g. the name of the file that the user retrieves, the file contents)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

ext.png Download



Q5.2 VPN traffic at internal router

1 Point

In the packet capture on the *external router*, which of the following can you read from the captured packets?

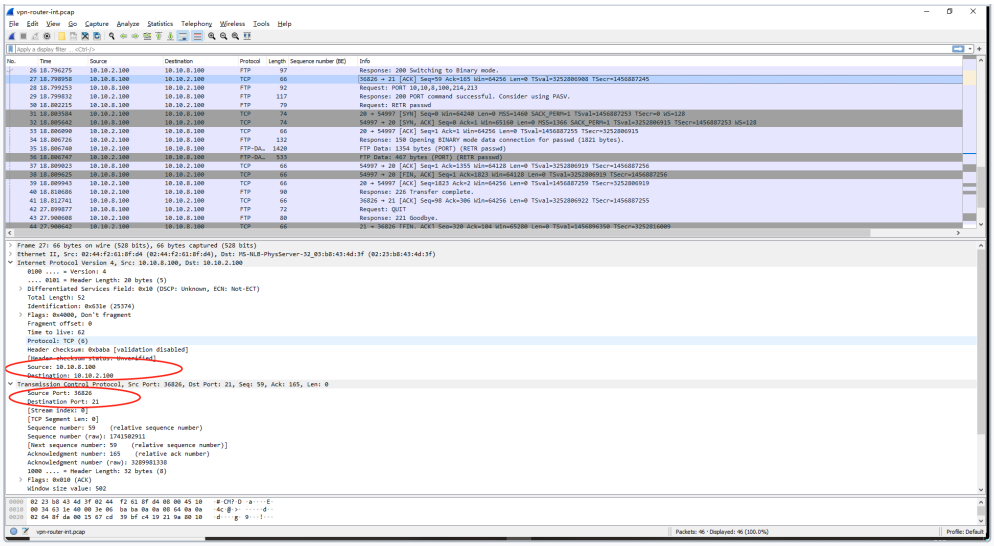
This information would be visible to a potential eavesdropper located somewhere along the network path **between the VPN server and the FTP server**.

Select all that apply:

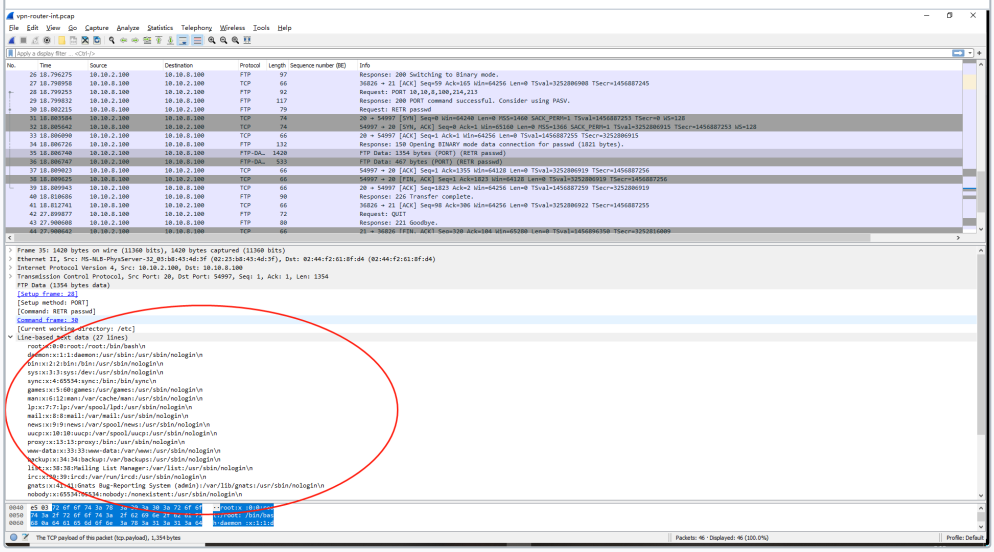
- [ ] IP address of "juliet" on the external network (not the address on the VPN tunnel!)
- [x] IP address of the VPN server, "vpn" (i.e. the fact that the client is using this particular VPN server)
- [x] IP address of the "server" node (i.e. the fact that the client is connecting to this particular FTP server)
- [ ] UDP port 1194 (i.e. this connection uses the well-known port number of OpenVPN, so eavesdroppers can identify it as VPN traffic)
- [x] TCP port 21 (i.e. this connection uses the well-known port number of FTP, so eavesdroppers can identify it as FTP traffic)
- [x] Session data (e.g. the name of the file that the user retrieves, the file contents)

To support your answer, upload screenshots from Wireshark or `tcpdump`. Annotate your screenshots to show *each* of the items above, and indicate whether you can read them (they are unencrypted) or not (they are encrypted).

▾ int.png Download



▾ int2.png Download



Q6 Firewalls (Section 9.12, Exercises 8 & 9)

1 Point

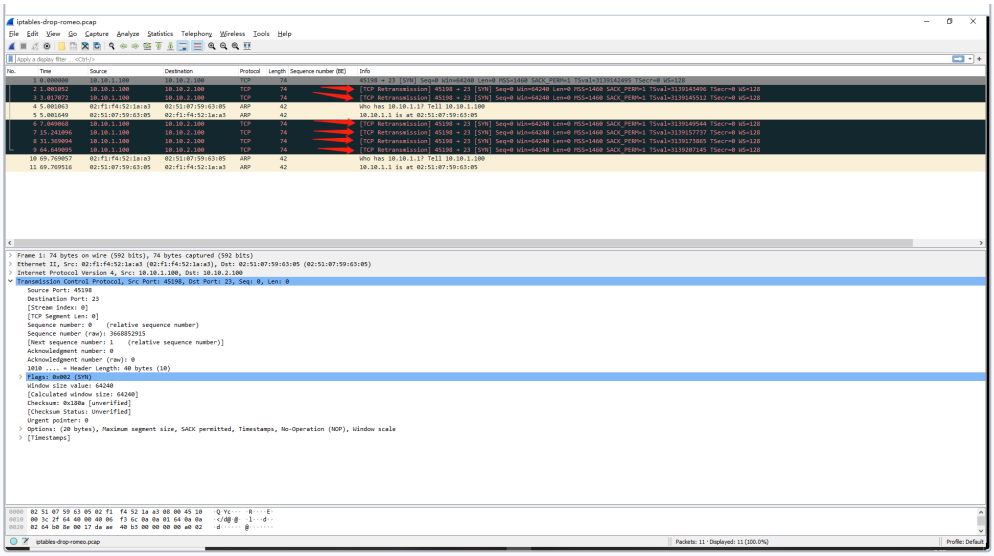
Q6.1 Firewall with DROP (Section 9.12, Exercise 8)

0.5 Points

Can you `telnet` to the "server" from "romeo"? Explain.

Use screenshots of your Wireshark or `tcpdump` output to show what happened. Annotate your screenshot: show where romeo attempts to initiate a `telnet` connection, and show what the response of the server is.

▾ drop.png Download



We can't telnet to the server from romeo since in server's we appe nd a rule to the end of the INPUT chain that drop all packet comin g to server. So the packer from romeo never get to server, after a f ew retransmission, romeo stop the connection and output connect ion timed out.

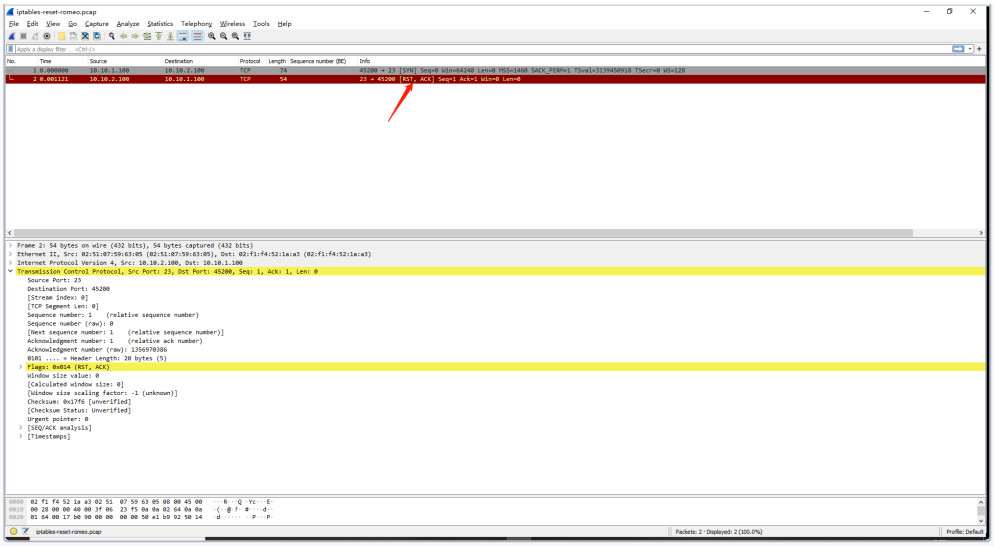
Q6.2 Firewall with RST (Section 9.12, Exercise 9)

0.5 Points

Can you `telnet` to the "server" from "romeo"? Explain the different between the `tcpdump` output in this exercise and the previous exercise.

Use screenshots of your Wireshark or `tcpdump` output to show what happened. Annotate your screenshot: show where romeo attempts to initiate a `telnet` connection, and show what the response of the server is.

▾ reset.png Download



In this case, instead of drop the packet. server will send back a RS T ACK and the connection will be refused.

Lab 9: Network management and Security

UNGRADED

STUDENT

Tingyu Yang

TOTAL POINTS

- / 10 pts

QUESTION 1

SNMP (Section 9.9) — 1 pt

1.1 SNMP community string (Section 9.9, Exercise 2) — 0.5 pts

1.2 Retrieving SNMP data (Section 9.9, Exercise 1) — 0.5 pts

QUESTION 2

Telnet and SSH (Section 9.10, Exercise 4) — 2 pts

2.1 telnet — 1 pt

2.2 SSH — 1 pt

QUESTION 3

FTP and SFTP (Section 9.10, Exercise 6) — 2 pts

3.1	FTP	1 pt
3.2	SFTP	1 pt

QUESTION 4

HTTP and HTTPS — 2 pts

4.1	HTTP	1 pt
4.2	HTTPS	1 pt

QUESTION 5

Network layer security — 2 pts

5.1	VPN traffic at external router	1 pt
5.2	VPN traffic at internal router	1 pt

QUESTION 6

Firewalls (Section 9.12, Exercises 8 & 9) — 1 pt

6.1	Firewall with DROP (Section 9.12, Exercise 8)	0.5 pts
6.2	Firewall with RST (Section 9.12, Exercise 9)	0.5 pts