Assignment #3 Solution

Part 1: Short Answer Questions

1) Consider the RSA algorithm to answer the following questions. Please provide STEP-BY-STEP details of the encryption and decryption.
   a) Encrypt the 4-bit message 1011 with p = 5, q = 13, e = 7.
   b) Find the corresponding d.
   c) Decrypt the ciphertext.

   We are given p=5 and q=13, so n=65. Choose e=7, and we get d=7 since e*d-1=48, 48 is divisible by (p-1)*(q-1). M=$(1011)_2$=11, so m**e= 19487171, and the c = (m**e) mod n = 41.
   To decipher, c**d = 194754273881, and m = (c**d) mod 65 = 11.

2) The Diffie–Hellman key exchange protocol is known to be vulnerable to a "man-in-the-middle" attack. Explain why? Outline how Diffie–Hellman can be extended to protect against this possibility.

   This vulnerability is present because Diffie-Hellman key exchange does not authenticate the participants. Possible solutions include the use of digital signature and other protocol variants.

3) Base64 encoding processes input blocks of size 24 bits at a time. How would it deal with input data whose length is not exactly 24 bits? Explain your answer.

   The base64 encoding actually defines 65 transmission characters; the 65th, "=", is used as a pad character. The data file is processed in input blocks of three bytes at a time; each input block translates to an output block of four 6-bit pieces in the base64 encoding process. If the final input block of the file contains one or two bytes, then zero-bits are first added to bring the data to a 6-bit boundary (if the final block is one byte, we add four zero bits; if the final block is two bytes, we add two zero bits). The two or three resulting 6-bit pieces are then encoded in the usual way, and two or one "=" characters are appended to bring the output block to the required four pieces. In other words, if the encoded file ends with a single =, then the original file size was ≡ 2 (mod 3); if the encoded file ends with two =s then the original file size was ≡ 1 (mod 3).

4) Using the browser of your choice, find out what certification authorities for HTTPS your browser is configured by default to trust. Do you trust these agencies? Find out what happens when you disable trust of some or all of these certification authorities.

5) Why do you think TCP is not used for streaming multimedia (audio, video) files? Explain in detail.

   Providing timely delivery is more crucial in multimedia applications than providing reliability. The multimedia applications can tolerate occasional losses without any need to

retransmit all the lost packets. With regards to congestion control, multimedia applications usually have their own application level mechanisms to handle congestion because it often requires understanding of the application data and TCP is not a suitable layer for providing such congestion control mechanisms.

6) Explain registration, agent discovery, tunneling and reverse tunneling in mobile IP.

Agent Discovery:
Agents advertise their presence by periodically broadcasting their agent advertisement messages. The mobile node receiving the agent advertisement messages observes whether the message is from its own home agent and determines whether it is in the home network or foreign network.
Agent Registration:
Mobile node after discovering the foreign agent, sends registration request (RREQ) to the foreign agent. Foreign agent in turn, sends the registration request to the home agent with the care-of-address. Home agent sends registration reply (RREP) to the foreign agent. Then it forwards the registration reply to the mobile node and completes the process of registration.
Tunneling:
It establishes a virtual pipe for the packets available between a tunnel entry and an endpoint. It is the process of sending a packet via a tunnel and it is achieved by a mechanism called encapsulation. It takes place to forward an IP datagram from the home agent to the care-of-address. Whenever home agent receives a packet from correspondent node, it encapsulates the packet with source address as home address and destination as care-of-address.

Part 2: Problem

1. A web page consisting of a HTML page (10 kB in size) and 10 JPEG images (each 50kB in size) is requested by a browser in a 10Mbps network where RTT between the browser and the server is 100ms. Ignoring any delay due to slow start and network congestion, compute the total HTTP response time for:

(a) persistent HTTP with pipelining

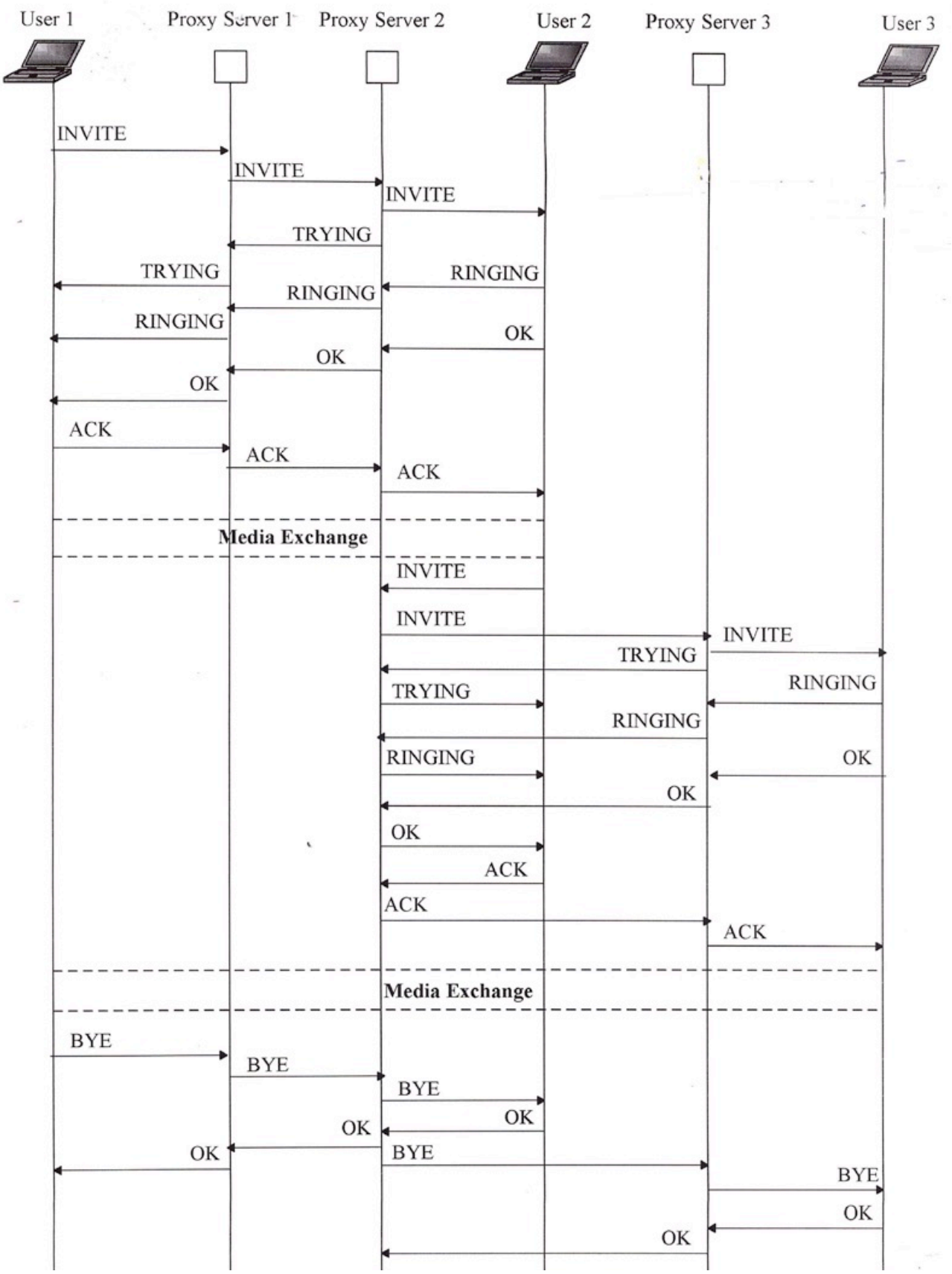
Total HTTP response time
= response time for HTML page + response times for the ten images
= 2 x RTT + transmission delay to request and receive base HTML file 2 x RTT transmission delay to request and receive base HTML file+ 1 x RTT + transmission delay to request and receive the 10 images
= 208ms + 100ms + 400ms = 0.708s

(b) non-persistent HTTP with no parallel connections

Total HTTP response time
= response time for HTML page + response times for the ten images
= connection setup (RTT) + request for HTML page (RTT)
= connection setup (RTT) + request for HTML page (RTT)+ transmission delay for HTML page+ 10 x (connection setup (RTT) + request for image (RTT) + transmission delay)

$=2 \times RTT + 10 \cdot 10^3 \times 8/10 \cdot 10^6 + 10 \times (2 \times RTT + 50 \cdot 10^3 \times 8/10 \cdot 10^6)$
$= 208ms + 10 \times (200ms + 40ms) = 208ms + 2400ms = 2.608s$

2. Three users, all belonging to different ISPs, plan to set up a conference call among themselves using SIP. First, User 1 connects with User 2 who then invites User 3 to join the conversation. You are asked to provide a timing diagram that shows all message exchanges until User 1 terminates the call.


User 1
Proxy Server 1
Proxy Server 2
User 2
Proxy Server 3
User 3
INVITE
INVITE
INVITE
TRYING
TRYING
RINGING
RINGING
RINGING
OK
OK
OK
ACK
ACK
ACK
Media Exchange
INVITE
INVITE
INVITE
TRYING
RINGING
TRYING
RINGING
RINGING
OK
OK
OK
ACK
ACK
ACK
Media Exchange
BYE
BYE
BYE
OK
OK
OK
BYE
BYE
OK
OK