CS5321

2011/2012 Semester 1

HW 1

Notes:

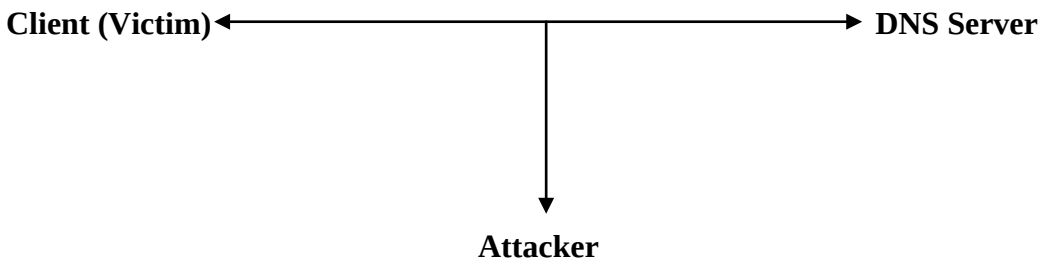
(1) Homework assignment should be done independently.

(2) Any queries regarding this assignment should be directed to Fai Cheong (TA) at faicheon@comp.nus.edu.sg

(3) Grading will be done individually with the TA and Lecturer. We will assign a timeslot (out of the 30 slots) for your demo. Please indicate which day or timeslots you are NOT available and send it to faicheon@comp.nus.edu.sg before October 1, 2011.

Days	6:30 – 7:00pm	7:00 – 7:30pm	7:30 – 8:00pm	8:00 – 8:30pm	8:30 – 9:00pm	9:00 - 9:30pm	9:30 – 10:00pm
10/10/2011 (Mon)	1	2	3	4	5	6	7
11/10/2011 (Tue)	8	9	10	11	12	13	14
12/10/2011 (Wed)	15	16	17	18	19	20	21
13/10/2011 (Thu)	22	23	24	25	26	27	28
14/10/2011 (Fri)					29	30	

Question 1



The client is surfing the web and DNS lookups are performed while surfing. Your task is to insert a fake DNS response to the client so that he would be surfing to another address.

Requirements:

1. Run your program using the following parameters:
   - *./program_name victim_ip ip_address_to_inject*

2. Listen and wait for Standard Query A before attempting to inject a fake DNS response.

3. The DNS response you inject MUST be accepted by the system such that the client would now be surfing to the "fake" address.

4. You can use the PCs that are connected through the hubs in the lab to eavesdrop the victim's packets (See Lab Setup). Alternatively, you can incorporate techniques in Question 2 to intercept victim's packets. However, by doing so would not gain more credits.

5. Show and explain your code and design during demo.

Assumptions:

1. You can assume default DNS UDP port 53.

2. You can assume there is only 1 query for each DNS request packet.

3. The query class is Internet address (IPv4).

Example

You can use nslookup command to do the DNS query at the client instead of using a browser. If you use a web browser, you may need to flush web/dns cache or restart your web browser if you query to the same site.

Attacker

[attacker@localhost ~]$ ./dnshijack 192.168.100.131 11.22.33.44

Client

[client@localhost ~]$ nslookup www.comp.nus.edu.sg

Server: 192.168.100.2
Address: 192.168.100.2#53

Name: www.comp.nus.edu.sg
Address: 11.22.33.44

} Your program should be able to change this to the injected ip address.

[client@localhost ~]$

If the victim is surfing using web browser and the attacker issue the command
[attacker@localhost ~]$ ./dnshijack 192.168.100.131 137.132.80.57

Then the victim would be surfing to NUS SoC (137.132.80.57) instead of the victim intended website.

Question 2

You (the attacker) have got access to a network port in the LAN. Perform ARP spoofing on a victim so that the victim traffic goes to you. To avoid suspicion, you should help to forward the victim traffic to the gateway so that victim surfing experience would not be affected.

Requirements:

1. Run your program using the following parameters:
   - *./program_name victim_ip*

2. You should use pcap to do packet forwarding rather than Linux IP forwarding mechanism in this assignment.

3. Show and explain your code and design during demo.

4. For simplicity, you are allowed to hardcode the Gateway IP and MAC address into your program.

Note:

1. You can check the ARP entries of victim by using the command: *arp -a*

Laboratory Setup:

1. The lab to be used for this homework is the Operating Systems and Security Lab, COM1-B-13. Linux machines are provided in the lab for this assignment.

2. Not all PCs in the lab are connected through switches. When you enter the lab, there are 2 columns of PCs on the left (about 12 PCs). These PCs are connected through hubs. If you do not want to perform ARP spoofing for question 1, you can use these PCs. The hub configuration allows you to sniff traffic without the need to perform ARP spoofing.

3. Each student will have his/her own account. Please lookup the file "login.xls" for your account. Your initial password is the login name. Do change your password upon your first login using the command "*passwd*". Contact TA if your password does not work or if you cannot find your name in "login.xls".

4. The lab is open 24/7, however on every Monday 1000-1400, Wednesday 15-1700, Thursday 1000-1400, Friday 0900-1200 and 1400-1500 the lab will be used by other course students. Please refrain from entering the lab during these hours.

5. You will use C/C++ and pcap library to do this assignment. However, as you do not have root access to the machines, the direct use of pcap library routines is prohibited. Instead, during compilation, you should link your program to the cs5321 library. An example of how you might compile your program in the lab is as follows:

   gcc -o cs5321q1 -l cs5321 cs5321q1.c

   If you are doing the assignment in your own PC using pcap library, an example of how you might compile your program is as follow:

   gcc -o cs5321q1 -l pcap cs5321q1.c

   (Use g++ if you are compiling C++ files - .cpp)

6. Note that the cs5321 library provides only a subset of the pcap library, it does not contain all the functions provided by pcap. Specifically, only the following functions are provided:

   - pcap_breakloop
   - pcap_close
   - pcap_compile
   - pcap_freecode
   - pcap_geterr
   - pcap_inject
   - pcap_lib_version
   - pcap_lookupdev
   - pcap_lookupnet

- pcap_loop
- pcap_open_live
- pcap_setfilter

Hence, if you are working on your own PC using pcap, make sure you only use the above functions. During demo, you will need to bring your code to the lab and compile it with cs5321 library. It is your responsibility to make sure your program works with the cs5321 library. Bear in mind that the cs5321 library is NOT an exact duplicate of these functions in pcap libary. One **important note** here is that the cs5321 library is currently not thread safe, hence do NOT use multiple threads or run multiple copies of your program at the same time.

7. To use wireshark in the lab, open a terminal window and execute the following command: sudo wireshark

For those who want to work and demo using their notebook:

We do allow students to do their assignments/demo on their notebook. However, you should take note of the following:

1. No technical assistance or whatsoever will be provided.

2. You are allowed to use OS other than Linux.

3. To be fair to other students, only pcap library is allowed.

Some Useful links:

You may find the following links useful:

- Packet Capture with libpcap
  http://www.tcpdump.org/pcap.htm

- Ethernet
  http://wiki.wireshark.org/Ethernet

- IP
  http://www.networksorcery.com/enp/protocol/ip.htm

- UDP
  http://www.networksorcery.com/enp/protocol/udp.htm

- ARP
  http://www.networksorcery.com/enp/protocol/arp.htm

- DNS
  http://www.networksorcery.com/enp/protocol/dns.htm

- Wireshark (this tool can be handy to see what was sent in the network)
  http://www.wireshark.org/