

Homework 2

Due: February 25, 2009, 11:59 PM EST

Please hand in your solutions as a pdf document

Problem 1 In July 2008, Dan Kaminsky gave a talk entitled “Black Ops 2008: It’s The End Of The Cache As We Know It” at the annual BlackHat conference (www.blackhat.com). The presentation is available at www.doxpara.com/DMK_BO2K8.ppt:

1. Read the presentation and understand Kaminsky’s cache poisoning attack. Describe how would you use this attack to spoof the domain name `cs.brown.edu`.
2. Explain why a large value for the TTL (time to live) of replies to DNS queries does not prevent Kaminsky’s attack.
3. Suppose now that the transaction ID (TXID) can take values from 1 to 65,536 and is randomly chosen for each DNS request. If the attacker sends 1,024 false replies per request, how many requests should he trigger to compromise the DNS cache of the victim with probability 99%?
4. Give three different techniques that the attacker can use to make the victim send DNS requests to the domains chosen by the attacker.
5. Propose modifications of the DNS architecture to defend against Kaminsky’s attack.

Problem 2 SYN flooding attacks used to be dangerous for servers because they could cause denial-of-service attacks. One of the proposed solutions to this problem is an approach called syn cookies. SYN cookies work by constructing a difficult-to-predict sequence number for the SYN-ACK handshake with the following components:

- (5 bits) A timestamp t modulo 32;
- (3 bits) The maximum segment size m that the server would have stored in the SYN queue entry;
- (24 bits) A hash of the server IP address and port number, the client IP address and port number, and the value t .

This sequence number is included in any future interaction, so it would NOT be possible to get another SYN queue entry from the same IP address. This approach seems pretty effective, but let’s discuss a potential weakness.

1. You are a system administrator at a large (think /16 set of IP addresses) network and you want to use SYN cookies to create a DOS attack on a web server. How do you accomplish this?
2. What throughput do you need to maintain the DOS?
3. How could you prevent a rogue sysadmin from doing this to you?

Problem 3 TCP is a reliable protocol with certain guarantees about service. It guarantees

that a stream of data will be sent from one host to another without duplication or lossiness. It accomplishes this using ACK or acknowledgement messages, which a client sends back to a server once it receives a packet. The server will use the arrival time of this ACK to estimate the round-trip time between the server and client. It uses this trip time to determine how many packets to transmit before requiring an ACK. The number of packets allowed to be outstanding before requiring an ACK is known as a congestion window. As a server receives ACKs it dynamically adjusts the congestion window to reflect the perceived greater or lesser bandwidth available.

An optimistic ACK attack is a denial of service, where an attacker tries to make a server increase its sending rate until it runs out of bandwidth and cannot effectively serve anyone else. The attack is accomplished by an attacker sending ACKs to packets before they have been received to make TCP increase its transmission speed. Every time it receives a packet the congestion window increases. Let's talk about some aspects of this problem and some of the proposed solutions to this problem.

1. The server uses additive increase/multiplicative decrease¹ to adjust the congestion window. It is calculated as follows:

$$w \leftarrow w - aw,$$

when loss is detected and

$$w \leftarrow w + b/w,$$

when an ACK arrives, where w is the current window and b and a are very small constants used to dictate the degree of adjustment. This exploit attempts to force a server to increase the size of its congestion window. How does the AIMD decrease algorithm make this still a time-consuming denial-of-service approach?

2. One solution proposes randomly not sending segments to a suspected receiver. Why and how would this work? What is the main disadvantage of this approach?
3. Another workaround proposed is throttling bandwidth to suspected users. What are the pros and cons to this approach? What type of attackers would this defend against?
4. If we went with the previous solution, what would be an effective way to decide what the bandwidth limit would be for a suspected client? Picking a hard coded limit is not a good option because clients at various distances have differing speed expectations.

Problem 4 For this homework assignment you will analyze the access control policies of a small directory on a windows virtual machine. You will be able to run our virtual machine only on computers in the iLab. On an iLab computer, run `vmplayer` in a command shell. Click on `Open an existing virtual machine`, and open the machine `WindowsXP.vmx` in the directory `/course/cs166/share/TrACE/WindowsXPi`, where i is a number between 0 and 9. If prompted that the machine you have selected is currently in use, select cancel and try again with a different value for i . If prompted to indicate whether the machine has been moved or copied, choose `I copied it`. After you finish the assignment, remember to shutdown the virtual machine from the start menu (i.e., don't just close `vmplayer`). Once you have logged in to one of the virtual machine as `Guest`, you will see the `Home` folder and the `Trace.exe` tool on the desktop. Now answer the following questions:

¹<http://en.wikipedia.org/wiki/AIMD>

1. Use only the standard windows tools to inspect file `Home/Charlie/My Documents/Records.txt`
Which users among Alice, Bob and Charlie have read access to the file? Which ones have write access to the file?
2. You have discovered that Alice is in fact, a spy sent from one of your competitors to discover the secrets to your success. You suspect that she has gained various levels of access to files within one of your employee's home directory. Use TrACE to inspect the contents of folder `Home/Charlie`. Are there any files that Alice can access? If so, list her access permissions on each of those files.
3. Use whichever tools you like to answer the following questions:
 - (a) Find all of the folders in `Home` directory tree on which there is a break of inheritance;
 - (b) Determine the access permissions for the pairs (user, file) in the table below and for each pair, list all the permissions (out of read, write and execute) that the user has to the file. Also, explain why the ACL of the file results in the access level that you indicated for the user.

User	File
Alice	<code>Home/Bob/Shared/BusinessContacts.txt</code>
Bob	<code>Home/Bob/Shared/BusinessContacts.txt</code>
Charlie	<code>Home/Charlie/MyDocuments/ExpenseReports/August.txt</code>

- (c) Which tool did you use? Why? How much time did it take?