CS 155 Final Exam

1. (*21 points*) ..... Short Answer

(a) (*3 points*) What aspect of the spam problem will DKIM prevent?

Answer: Phishing email pretending to be from bank. Bounce messages to hotmail.

(b) (*3 points*) How are secret salts used in managing a password file? What is their purpose?

Answer: Add artificial entropy to user passwords. Host verifies password with all possible values of the secret salt.

(c) (*3 points*) Suppose trusted computing (TCG) is implemented without extra hardware. That is, suppose all hardware secret keys are stored in the OS kernel instead of a TCG chip. The OS can use these keys to attest to application code running on top of it. Are such attestations trustworthy? If so, explain why. If not, describe an attack.

Answer: No. Memory snapshots using a VMM can be used to extract the secret keys.

(d) (*3 points*) What is the same-origin principle? Say briefly how this policy is or is not applied to (i) cookies, (ii) visited links, and (iii) frames.

Answer: A web site should only have access to parts of the browser state set by the web site. Only cookies set by a web site are visible to the site. Unfortunately, history entries are visible to all sites.

(e) (*3 points*) What is the "confused deputy" problem? How does Java stack inspection help reduce this general problem?

Answer: Trusted sub system (e.g. font loader) accesses a restricted object due to a request from an untrusted sub system.

(f) (*3 points*) What is the principle of least privilege? Why is it important?

Answer: Only grant a section of code the privileges it needs to do its job. For example, if a process does not need write access to a file system, to not give it write privileges. This prevents attacks on the code (such as a buffer overflow) from doing malicious writes.

(g) (*3 points*) In comparison with access control lists, what are the advantages and disadvantages of capabilities? List one advantage and one disadvantage.

Answer: Access control lists make it easier to revoke a specific permission of a specific user. Capabilities allow a user to pass a permission to another user.

2. (*10 points*) ..... Electronic Voting

In a typical U.S. election, voting machines are purchased by a local election board from a supplier. Before each election, election board employees configure each machine for the upcoming election so that each machine will present the correct list of candidates and other voting options. During the election, voters come to each polling place, identify themselves to voting officials, and obtain a ballot or card to place in a machine. Each voter inserts their ballot or card, marks their votes in some way, and removes the ballot or card. After voting, the voter places the removed ballot, card, or any printout from the machine in a box used for this purpose.

After votes are cast, votes can be counted either using votes stored (or electronically transmitted) by each machine, or by using a marked ballot, card, or printout produced when a voter completes a vote. When a vote is contested, a recount is done in whatever way the voting technology allows.

(a) (*2 points*) Consider an electronic system, where voting machines store a vote count that is read from the machine at the end of the election, and no card, ballot, or machine printout records the vote. For each part of the system – the voting machine, the election board employees, and the voter – explain what this part of the system is trusted to perform.

Answer: Voting machine: trusted to correctly display ballot, record votes, and count votes. Election board employee: trusted to correctly enter the correct ballot and not corrupt the machine, Voter: untrusted.

(b) (*2 points*) What characteristics of this system prevents a single voter from voting twice?

Answer: The media (ballot or card) removed from the machine is marked so that it cannot be placed in another machine to cast a second vote. The voting machine is trusted to record only one vote per ballot or card.

(c) (*2 points*) How are voters prevented from proving how they voted to someone else outside the polling place? Why is this considered important?

Answer: Voters must leave their ballot or card in the vote box. The cannot leave with any proof of their vote. This is important because it prevents vote selling.

(d) (*2 points*) Some systems provide a printout that can be read (and checked for correctness) by the voter before is it placed in a collection box. How does this reduce the "trusted computing base" of the voting system?

Answer: A voter-verifiable audit trail makes it possible to count votes independently, or perform an independent recount. This eliminates the need to trust the voting machine to count the votes correctly.

(e) (*2 points*) Consider the possibility of Internet voting, in which voters use their browsers to vote at a voting web site. Assume that each voter is given a password, and disregard risks associated with password authentication. Explain why at least two goals of the voting process are difficult or impossible to achieve in this scenario.

Answer: Vote buying: a voter can sell his/her password. Voter assurance: voters have no reason to believe that their votes are counted, or counted properly.

3. (13 points) ..... Buffer overflows

Consider the following C code fragment:

```
int func(int i, double *data1, double data2) {
  double *p = data1;
  double *vec[10];
  if ((i<0) || (i>10)) return;
  vec[i] = data1;
  *p = data2;
}
```

(a) (5 points) Explain why this code is vulnerable to a control hijacking attack. Briefly explain how your attack works.

Answer: Send $i = 10$ and $data1 = RetAddr$

(b) (3 points) If this code is compiled with Stackguard, will the overflow attack be prevented? If so explain why, if not explain why not.

Answer: No.

(c) (3 points) If this code is compiled with Stackshield, will the overflow attack be prevented? If so explain why, if not explain why not.

Answer: Yes.

(d) (2 points) If this code is run with libsafe, will the overflow attack be prevented? If so explain why, if not explain why not.

Answer: No.

4. (13 points) ..... Kerberos Authentication

Kerberos involves three two-message exchanges, one between the client and the *Key Distribution Center (KDC),* one between the client and the *Ticket Granting Service (TGS),* and one between the client and the *server (S)* chosen by the client.

In Kerberos v4, the initial communication between the client C and the KDC D goes like this:

1. C sends a ticket request containing C's name and a TGS's name T.
2. The KDC checks that both C and T are known to the system.
3. The KDC creates a ticket containing C's and T's names, C's network address, the current time, the lifetime of the ticket, and a session key K_{CT}. This ticket is encrypted with T's secret key K_{DT} known to both the key-distribution center D and the ticket-granting service T.
4. The reply to C consists of the ticket just described, T's name, the current time, the lifetime of the ticket, and the session key, all encrypted with C's secret key K_C. To keep messages that are intended for one purpose from being mistakenly used for another, the plaintext of the encrypted reply contains a constant string "krbtgt" identifying this as a ticket-granting ticket.
5. The client decrypts the reply and saves the ticket for use.

Questions:

(a) (*3 points*) Explain briefly, in general terms, the purpose of the each of the three exchanges (between the client and KDC, client and TGS, and client and S).

Answer: The KDC recognizes the user's password and grants a ticket that allows the user to request access to servers. The TGS lets the client ask for access to several servers, and servers provide some (useful) service.

(b) (*2 points*) Assume that the user's password is not stored on the client machine, and the client's key K_C is computed from the user's password by a known function. Why is Kerberos more convenient, for the human user, than a system in which the TGS is eliminated, and the client makes a Kerberos-style request to the KDC for each server connection?

Answer: The user only needs to enter her password once for each KDC ticket lifetime.

(c) (*2 points*) In Kerberos v4, it is possible for an attacker to request a ticket for C, or simply overhear a request and response for C. Explain how this allows an attacker to do an offline dictionary attack.

Answer: The attacker can intercept the KDC response, try candidate passwords, and recognize success with the string "krbtgt" appears.

(d) (*3 points*) In Kerberos v5, a *nonce,* or random number, is added to the client's request to the KDC, and included (as part of the encrypted response) in the reply from the KDC. Nonces are similarly used in the request and response from the TGS. What purpose does this serve?

Answer: Eliminate certain replay attacks.

(e) (*3 points*) A Kerberos *realm* consists of a KDC, a TGS, a number of clients sharing keys with the KDC, and a number of application servers sharing keys with the TGS. In cross-realm authentication, a client in one realm wishes to use a server in another realm. Explain briefly how Kerberos is used in cross-realm authentication (across two realms) and state what key(s) must be shared between the two realms.

Answer: In cross-realm Kerberos, a TGS in the realm of the client can supply a ticket for the TGS in the second realm. For this to work, the two TGS's must have a shared secret key.

5. (*15 points*) ..... SQL Injection

In class we discussed the following PHP script for a login page:

```
$username = $_GET[user];
$password = $_GET[pwd];

$sql = "SELECT *
        FROM   usertable
        WHERE  username = '$username'
        AND    password = '$password' ";
$result = $db->query($sql);
if ($result->num_rows > 0) { /* Success */ }
else                       { /* Failure */ }
```

(a) (*2 points*) Explain why a URL where `user` is set to `" ' or 1 = 1 -- "` will result in a successful login.

Answer: The first quote closes supplied parameter, the 1=1 always evaluates to true, and the "- -" causes the rest of the line to be ignored.

(b) (*2 points*) Suppose we change lines 1 and 2 to

```
$username = addslashes($_GET[user])
$password = addslashes($_GET[pwd])
```

Recall that the `addslashes` function adds a slash before every quote. That is `addslashes("a'b")` will output the string `"a\'b"`. Explain why this prevents the attack from part (a).

Answer: The quote in `$username` will no longer have any effect.

(c) (*9 points*) Does `addslashes` completely solve the problem? Consider the GBK Chinese unicode character set. Some characters in GBK are single bytes while others are double bytes. In particular, the following table shows a few GBK characters:

```
0x 5c     =  \
0x 27     =  '
0x bf 27  =  ¿'
0x bf 5c  =  縗
```

That is, the database interprets `0xbf27` as two characters, but interprets `0xbf5c` as a single chinese character.

Show that using `addslashes` as in part (b) leads to a SQL injection attack. What value of `user` will result in a successful login?

Answer: `user = " 0x bf 27 or 1=1 - - "`

(d) (*2 points*) How should `addslashes` be implemented to defend against your attack from part (c)?

Answer:

6. (*10 points*) ..... Stealth port scanning

Recall that the IP packet header contains a 16-bit *identification* field that is used for assembling packet fragments. IP mandates that the identification field be unique for each packet for a given (SourceIP,DestIP) pair. A common method for implementing the identification field is to maintain a single counter that is incremented by one for every packet sent. The current value of the counter is embedded in each outgoing packet. Since this counter is used for all connections to the host we say that the host implements a *global* identification field.

(a) (*2 points*) Suppose a host P (whom we'll call the Patsy for reasons that become clear later) implements a global identification field. Suppose further that P responds to ICMP ping requests. You control some other host A. How can you test if P sent a packet to anyone (other than A) within a certain one minute window? You are allowed to send your own packets to P.

Answer: Send a ping to P at the beginning of the window and another ping at the end of the window. Check if the difference in identification values is greater than 1.

(b) (*5 points*) Your goal now is to test whether a victim host V is running a server that accepts connection to port n (that is, test if V is listening to port n). You wish to hide the identity of your machine A. Hence, A cannot directly send a packet to

V, unless that packet contains a spoofed source IP address. Explain how to use the patsy host P to do this.
Hint: Recall the following facts about TCP:

- A host that receives a SYN packet to an open port n sends back a SYN/ACK response to the source IP.
- A host that receives a SYN packet to a closed port n sends back a RST packet to the source IP.
- A host that receives a SYN/ACK packet that it is not expecting sends back a RST packet to the source IP.
- A host that receives a RST packet sends back no response.

Answer: Send a SYN to V with src address set to P. Then test if P sent out a packet.

(c) (*3 points*) How would you change host P to avoid this problem? You are not allowed to modify the TCP/IP protocol or the services running on P. You may only modify the *implementation* of TCP/IP on host P.
Answer: Send $E_k(counter)$ in every packet.

7. (*10 points*) ..... Blue Security

An anti-spam company called Blue Security Inc. used a vigilante approach to fighting spam. Blue Security customers reported their spam to Blue Security, which analyzed it and sent back a set of instructions to a Blue Frog client running on the customer's machine. The client software used these instructions to visit the websites advertised by the spam messages and leave complaints on those websites. For each spam a user received, the Blue Frog client would leave one generic complaint. Blue Security operated on the assumption that as the community grew, the flow of complaints from hundreds of thousands of computers would apply enough pressure on spammers and their clients to convince them to stop spamming. A similar idea is the basis of an open source P2P system called Okopipi.

On May 1st 2006, Blue Security's web site came under a massive DDoS attack using a variety of techniques including *DNS amplification*. Subsequently, the company shut down.

(a) (*3 points*) How does a DNS amplification DDoS attack work?
Hint: Recall that a 60-bytes UDP query to a (recursive) DNS server can result in a 512-byte UDP response (or 4000-bytes with EDNS) to the source IP.
Answer: A reflector attack — send lots of DNS queries with the victim's source IP

(b) (*3 points*) What are some solutions to DNS amplification?
Answer: Prolexic-like proxy. Throttling at DNS server. Disable recursive queries from outside the domain.

(c) (*2 points*) Suppose Blue Security was still in business. Can the Blue Security service itself be used to mount a DoS attack? If so, explain how.
Answer: Yes. Send fake spam, with the DoS victim URL in the spam email.

(d) (*2 points*) Would SPF or DKIM prevent the exploit you described in part (c)?
Answer: No.

8. (*8 points*) ..... Firewalls

(a) (*2 points*) Explain the difference between packet filters and application layer proxies.

Answer: Packet filters look at packets one at a time, while application-layer proxies reconstruct application layer entities, such as email messages, files, and web pages.

(b) (*2 points*) Can a stateless firewall block TCP connection initiation requests from an external location to any local host, but at the same time allow returning traffic from connections initiated by local hosts? Why or why not?

Answer: Yes. The firewall filters out SYN-packets to a local host, but allows SYN-ACK and other packets to flow through.

(c) (*2 points*) What is the main security benefit of NAT and why is it useful to combine NAT with a firewall, instead of using separate NAT and firewall devices?

Answer: NAT hides the addresses of devices behind the NAT device an prevents attacks that use knowledge of internal network addresses behind the NAT device. Some firewall policies, such as allowing traffic to high-numbered ports only if there was a matching outgoing request, require port numbers and internal addresses. This is easier to determine the firewall also knows the NAT translation table.

(d) (*2 points*) In a distributed firewall, an administrator ships out firewall rules to hosts over an authenticated channel, and each host enforces its own policy. Give one advantage and one disadvantage of a distributed firewall, in comparison with a centralized firewall.

Answer: Advantage: Can filter traffic between internal hosts on the local network. For example, prevent ssh connections from certain internal hosts, avoiding possible attacks if they are compromised. Disadvantage: Cannot protect against external flooding of an internal network – in a DoS attack, the links between local hosts will be flooded, whereas this could be prevented by throttling incoming traffic at a gateway firewall.