Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.893 Fall 2009

Quiz II

All problems are open-ended questions. In order to receive credit you must answer the question as precisely as possible. You have 80 minutes to finish this quiz.

Write your name on this cover sheet.

Some questions may be harder than others. Read them all through first and attack them in the order that allows you to make the most progress. If you find a question ambiguous, be sure to write down any assumptions you make. Be neat. If we can't understand your answer, we can't give you credit!

THIS IS AN OPEN BOOK, OPEN NOTES EXAM.

Please do not write in the boxes below.

I (xx/10)	**II (xx/30)**	**III (xx/10)**	**IV (xx/10)**
V (xx/10)	**VI (xx/20)**	**VII (xx/10)**	**Total (xx/100)**

Name: Solutions.

I KeyKOS

1. [10 points]: Bob is running the privilege-separated Zoobar web site on a KeyNIX system, using code from lab 3. Suggest a way in which Bob can modify the Zoobar server-side code to take advantage of KeyKOS capabilities to improve the security of his site, in a way that he wouldn't be able to do on Linux.

Answer: Two important advantages of KeyKOS are that a process doesn't have to be root to do various things (e.g. for zookld to create a service in a chroot jail, by only granting it a capability to that specific directory, and not granting a capability to the top-level root directory), and that a process can protect itself from the Unix root user (e.g. by creating a separate KeyNIX universe, which will be secure even if the root account in the original KeyNIX universe is compromised.)

II Network protocols

Bob logs into an Athena workstation, which uses Kerberos to obtain a ticket for bob@ATHENA.MIT.EDU, and then runs Bob's mail client, which contacts Bob's post office server to fetch new messages.

2. [10 points]: Alice doesn't want Bob to know about an upcoming event, which was announced to Bob via email. To this end, Alice plans to intercept Bob's communication with his post office server, and to pretend that Bob has no new mail. Alice can observe and modify all network packets sent by Bob. How does Kerberos prevent Alice from impersonating Bob's mail server? Be as specific as possible; explain how Bob can tell between Alice and the real mail server in terms of network packets.

Answer: The server has to send back $\{timestamp + 1\}K_{c,s}$, which Alice cannot forge.

Now, Alice wants to read Bob's email, and intercepts all network packets ever sent and received by Bob's workstation (which is the only computer that Bob uses). However, Alice does not know Bob's password to access Bob's post office server, and Bob's packets to and from the post office server are protected by Kerberos.

3. [10 points]: Suppose that after Bob reads and deletes all of his mail, Alice learns what Bob's password was. Describe how Alice can obtain Bob's past messages.

Answer: Alice can use the password to decrypt Bob's TGT (which she captured in the past), then use the key in the TGT to decrypt Bob's service tickets, and then use the session key in the service ticket to decrypt Bob's mail traffic.

4. [10 points]: To prevent Alice from reading any more messages, Bob ensures that Alice cannot intercept any subsequent network traffic, and changes his Kerberos password. Could Alice still read Bob's mail after this? Explain why not or explain how.

Answer: Alice can continue to read Bob's mail (by connecting to Bob's post office server) until her ticket for Bob's principal expires. After that point, she will not be able to connect to the post office server as Bob.

Also, she can exploit slow slave replication and obtain fresh tickets for Bob's principal from a slave KDC.

III ForceHTTPS

5. [10 points]: Bob is developing a new web site, and wants to avoid the problems described in the ForceHTTPS paper. He uses HTTPS for all of his pages and marks all of his cookies "Secure". Assuming Bob made no mistakes, is there any reason for Bob's users to install the ForceHTTPS plugin and enable it for Bob's site? Explain why or why not.

Answer: ForceHTTPS will prevent users from accidentally accepting an invalid certificate for Bob's site (caused by an active attacker trying to impersonate Bob's web server).

ForceHTTPS will also prevent users from making HTTP accesses to Bob's site, but that in itself isn't a problem if the cookie is marked "Secure."

IV BitLocker

6. [10 points]: Alice wants to make BitLocker run faster. She decides that computing a different IV_s for each sector (pg. 13 in the BitLocker paper) is needlessly expensive, and replaces it with the fixed value $E(K_{\text{AES}}, e(0))$ instead. Explain how an attacker may be able to leverage this change to obtain data from a stolen laptop that uses BitLocker in TPM-only mode.

Answer:

BitLocker's sector encryption algorithm is still sector-specific—namely, the sector plaintext is XORed with a sector key K_s, as shown in Figure 1 on page 13 and described in Section 4.3 on page 14. However, since the AES-CBC key is the same for each sector, an attacker may be able to compute the XOR of two sector keys, by swapping the two encrypted sectors on disk. If the attacker swaps encrypted sectors s_1 and s_2, whose original plaintexts were p_1 and p_2, then the new decryption of s_2 will be $p_2' = p_1 \oplus K_{s_1} \oplus K_{s_2}$. If the attacker knows p_1, and can read the new value p_2', then he can deduce $K_{s_1} \oplus K_{s_2}$. At this point, if the attacker wants to place malicious data m_1 in sector s_1, he can place the value $m_1 \oplus K_{s_1} \oplus K_{s_2}$ in sector s_2 and swap the two sectors again. To be able to set a particular range of bytes in some sector to a given value, the attacker must be able to read and write the same range of byte offsets in some other sector inside the OS (e.g. being able to read and write a file is sufficient).

Thanks to Stephen Woodrow for pointing out a problem with our previous solution.

V Tor

7. [10 points]: Bob is running a hidden service on top of Tor, and wants to know how frequently he should choose new introduction points. Bob cares about his identity not being exposed, and about the availability of his service. Help Bob make an informed choice by explaining the costs and benefits of rotating introduction points either more or less frequently.

Answer: Rotating introduction points more frequently helps avoid DoS attacks on a fixed set of introduction points. Rotation also helps prevent a single introduction point from gaining long-term statistics on how often the service is accessed. Rotation does not improve Bob's anonymity, because Bob can keep building new circuits to the same introduction point. More frequent rotation places additional load on directory services that provide lookup functionality. However, this does not compromise anonymity either, since lookups and updates happen via anonymous Tor circuits as well.

VI BackTracker

8. [10 points]: Alice is using the VM-based BackTracker to analyze a compromised server, where an attacker obtained some user's password, logged in via SSH, exploited a buffer overflow in a setuid-root program to gain root access, and trojaned the login binary, all using high-control events that are still stored in the event logger. How can Alice figure out what *specific* vulnerability in the setuid-root program the attacker exploited? In what situations would this be possible or not possible?

Answer: If the vulnerability is triggered by command-line arguments, then Alice can figure out the vulnerability by looking at the arguments used when invoking the setuid program.

On the other hand, if the attack involves supplying specific inputs to the setuid program, and the attacker does so by hand, or by using a downloaded program, Alice might not be able to figure out what specific bug was exploited. (She would need some replay mechanism, either at the level of the entire VM, or at the network level, assuming the attacker did not encrypt the traffic.)

9. [10 points]: The VM-based BackTracker system requires no modifications to the guest OS, but nonetheless makes assumptions about the guest OS. List assumptions that are critical to back-tracking attacks that used high-control events.

Answer: Backtracker assumes that it can observe all system calls (i.e. it depends on a specific system call mechanism), that it can observe system call arguments, and that it can access kernel data structures for things like process IDs, inodes, etc. Backtracker also assumes the guest kernel is not compromised; one could think of this as a special case of the attacker changing the system call format.

VII 6.893

We’d like to hear your opinions about 6.893, so please answer the following questions. (Any answer, except no answer, will receive full credit.)

10. [10 points]: If you could change one thing in 6.893, what would it be?

Answer: Homework questions for each paper. More labs. . . .

End of Quiz