Homework 4

Jake Maschoff
U0581770

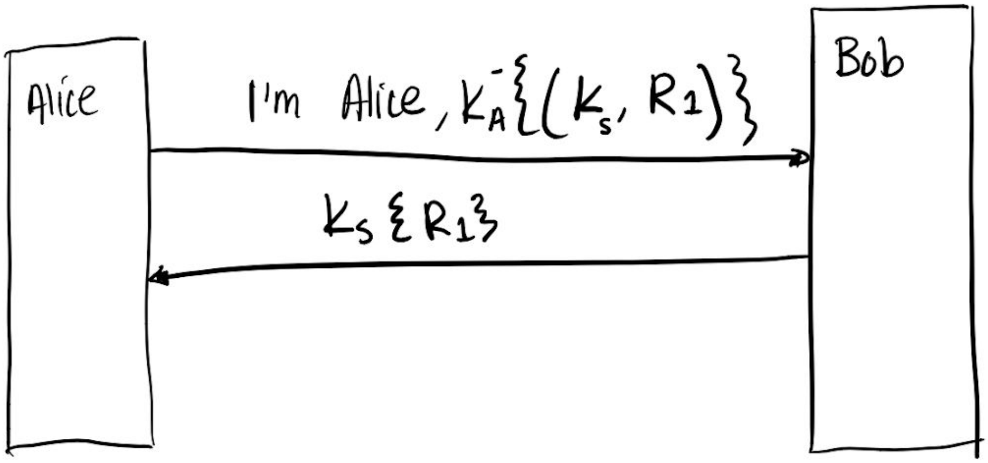
1. Password-Based Authentification
   a. Problem 2, Chapter 9, page 236.
      i. **Not against database reading. That is because Bob has Z stored, which is the hash of Alice's password. That means anyone could overrun Bob and take Z. With Z, you would be able to impersonate Alice with hash(Y, R) since Y = Z. This protocol is safe against eavesdropping, though.**
   b. Let a dictionary have 4096 words. Let a user pick 5 words at random for choosing a password (i.e., the password comprises of these 5 words). What is the strength of this password in bits? [Hint: Recall the discussion we had on an xkcd comic in the class.] Explain your answer.
      i. **Since the dictionary has 4096 words, that can be represented as 12-bits. If there are 5 words for each password, that means you would have 5 * 12-bits for your password. Thus giving you 60-bits of protection.**
2. Security-handshake Pitfalls
   a. Problem 2, page 288, chapter 11.
      i. **No, this protocol is not susceptible to a reflection attack. That is because Bob, the server, does the initiation of the authentification and thus picks the nonce value.**
   b. Problem 3, page 288, chapter 11.
      i. **$\{R+A\}_A$ and $\{R\}_{R+A}$ are both good session keys. That is because they will be different for each session, are unguessable by an eavesdropper, and don't consist of any encrypted value sent in previous messages in the protocol.**
   c. Consider the one-way authentication protocol shown in the figure below. Here, Bob is a stateless server, and therefore it is inconvenient to require him to remember the challenge he sent to Alice. Let KAB be the secret key shared between Alice and Bob. Now, this protocol is vulnerable to a replay attack where an eavesdropper can record R, KAB{R} pair and replay that later. If we enhance the protocol such that R represents the current time, is the protocol secure? Identify one strength and one weakness of this enhanced protocol.
      i. **With making R the timestamp, Bob could make it so they only accept the authentification of R within a certain time period. So if the authentification is sent 15 seconds after Bob's current system time, Bob would disallow this authentification because it would most likely be a replay attack. The weakness, though, is that if Trudy were able to send R, KAB{R} within that time period they would be authenticated as Alice.**

d. Problem 7, page 289, chapter 11.

   i. **If the system that Bob is using, uses microseconds, then there would be 1.2 billion different combinations of time stamps. This would make it infeasible for an adversary to try and reproduce K_{AB} with 1.2 billion values over 10 minutes.**

e. Problem 8, page 289, chapter 11.

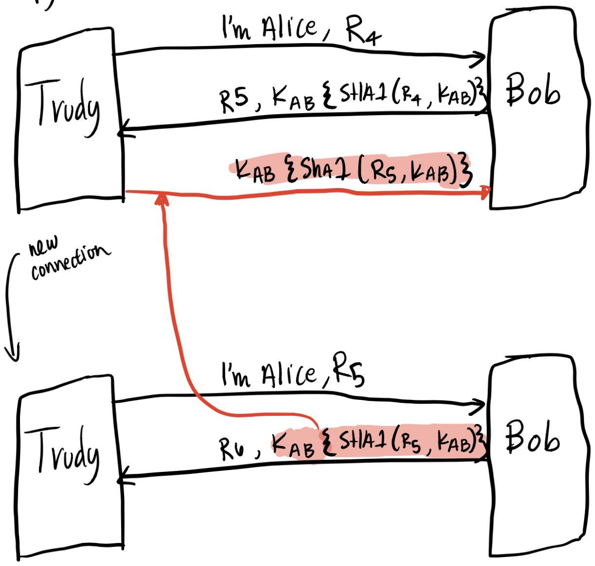
e)



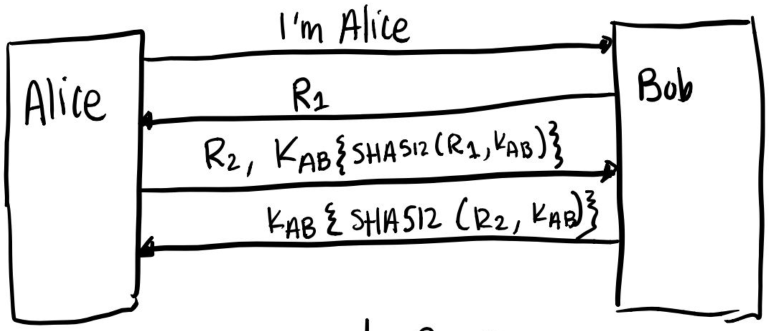
Alice picks some session key K_s, that they send to Bob with a nonce value. These 2 values are signed by Alice's private key. Bob is then able to decrypt this value via Alice's public key and grab K_s and R_1. Bob then sends back $K_s\{R_1\}$ so Alice knows Bob got the correct K_s and was able to decrypt their message to correctly send back R_1. Bob is sure it's Alice since it was signed by Alice.

f. With the help of figures and message exchanges, show the reflection attack on the following protocol: Here, KAB is the shared secret between Alice and Bob, R1 and R2 are random nonces (random numbers used only once), and the function f is a secret key function or a hash function. The protocol of this question is vulnerable to an offline dictionary attack by Trudy impersonating as Alice. Modify this protocol appropriately to remove this vulnerability.

f)



I'm Alice, R_4
Trudy
$R5, K_{AB}\{SHA1(R_4, K_{AB})\}$
Bob
$K_{AB}\{Sha1(R_5, K_{AB})\}$
new connection
I'm Alice, R_5
Trudy
$R_6, K_{AB}\{SHA1(R_5, K_{AB})\}$
Bob



I'm Alice
Alice
Bob
R_1
$R_2, K_{AB}\{SHA512(R_1, K_{AB})\}$
$K_{AB}\{SHA512(R_2, K_{AB})\}$

where R_1 and R_2 are nonces taken from a large sample space.

With taking protocol 11-11, it is no longer vulnerable to reflection attacks. Along with upgrading SHA1 to SHA512. This increases the search space from 2^{160} to 2^{512} which is much higher and harder to break.

3. Reading

a. Let us plot the observed clock offset, in microseconds, on the y-axis and the time since the start of the fingerprinting measurements, in seconds, at the fingerprinter, on the x-axis. Let (6, 60) and (7.5, 80) be two points at times 6s and 7.5s, where the clock offset is observed by the fingerprinter to be 60 and 80, respectively. Estimate the clock skew of the fingerprintee from these two points. You can assume that the network delays are negligible.
   i. **The clock skew is the slope of the two points, so that would be (80-60) / (7.5-6) = 20/1.5 = 40/3 secs/frame**

b. What clock skew-behavior do the authors observe in the case of virtual access points?
   i. They found that all the virtual APs being emulated on the hardware had the same clock skew. That is because each of these virtual APs don't implement their own virtual clock. They all use the same real hardware clock.

c. Why is it not easy to fabricate clock skews of access points?
   i. It is difficult to fabricate clock skews because the adversary is able to produce the beacon packets at any time to produce the wanted timestamps. However, the adversary is unable to know exactly when the actual AP is going to send their beacon packets. Because of that, their skew will be affected as to when the actual AP sends their beacon packets, and cause their skew to not be as predicted. Thus resulting in them being detected.

d. How do the authors address the problem of change in clock skew due to change in temperature?
   i. They propose a 'rolling signature' scheme. They do this by comparing the new current skew to the previously recorded clock skew. If that is less than the max skew variance, you then save the new current skew as the previously recorded clock skew to measure against the next recorded clock skew.

e. Would you expect the clock skew of an access point measured by one laptop to be the same as that measured by another laptop? Explain briefly.
   i. Yes, depending on their distance from the AP, their clock skews could vary. Let us assume laptop A receives frame$_1$ at t = 10 and frame$_2$ at t = 20. That would lead to a clock skew time of 10 sec/frame. Whereas for laptop B, they receive their frames later since they are farther away from the AP. So they receive frame$_1$ at t = 15 and frame$_2$ at t = 30. Thus leading to a clock skew time of 15 sec/frame for laptop B.