CS 798

Assignment 1

1. Setting up an SSL session, as described in Section 1.1.2, incorporates server-to-client authentication but no client-to-server authentication. Briefly discuss any reasons you can think of why the protocol may only use one-way authentication.

   Answer: The primary reason that SSL does not require client-to-server authentication is that most clients will not have certificates that are signed by a certification authority. Therefore the protocol is designed to work in the absence of client certiciates.

   Eventually, in most applications, it will be required for the client to present some kind of authenticating information to the server (e.g., a valid credit card number or userID and password), but this is usually done in the subsequent session by encrypting the client's information using the SSL session key, and not during the SSL protocol itself.

   Finally, note that the server's authenticated key is enough to allow the client and server to obtain a secret session key (though the technique used in SSL, namely, having the client choose a random master secret and send it to the server encrypted with the server's public key, is not very secure).

2. Consider the Rabin cryptosystem with $n = 638160384373$. Suppose the ciphertext is 377838732432. Factor n, and use the Rabin decryption algorithm to determin the four possible plaintexts whose encryption is n. You can implement the Chinese remainder theorem with two moduli yourself.

   Answer: The four possible plaintexts are 540886367662, 636241624978, 1918759395 and 97274016711.

3. The integer $p = 90872345251$ is prime.

   (a) Factor $p - 1$, and then verify that 2 is a primitive element in $\mathbb{Z}_p^*$ using the efficient test described in class. Show that 3 is not a primitive element.

   Answer: $p - 1 = 2^1 3^2 5^3 40387709^1$.
   $2^{(p-1)/2} \equiv 90872345250 \pmod{p}$, $2^{(p-1)/3} \equiv 44551634580 \pmod{p}$, $2^{(p-1)/5} \equiv 60603626420 \pmod{p}$, and $2^{(p-1)/40387709} \equiv 45008857185 \pmod{p}$. Hence, 2 is a primitive element modulo p.
   $3^{(p-1)/3} \equiv 1 \pmod{p}$, so 3 is not a primitive element modulo p.

   (b) How many primitive elements are there in $\mathbb{Z}_p^*$ for the given value of p. (Use the formula given in class!)

   Answer: $p - 1 = 2^1 3^2 5^3 40387709^1$. The number of primitive elements is

   $$\phi(p-1) = (2-1)(9-3)(125-25)(40387709-1) = 24232624800.$$

4. Prove that the Rabin cryptosystem is *insecure* in a chosen-ciphertext attack.

   Hint: Consider Algorithm 1.9.

   Answer: The following algorithm is a modification of Algorithm 1.9, in which the oracle RabinDecrypt is replaced by a decryption request (which is permitted in the chosen-ciphertext attack model):

   (a) choose a random integer $r \in \mathbb{Z}_n^*$
   (b) compute $y = r^2 \bmod n$
   (c) request a decryption of y, say x
   (d) if $x \equiv \pm r \pmod{n}$ then try again
   (e) otherwise, compute $p = \gcd(x + r, n)$ and $q = n/p$

   By using the same analysis as was done for Algorithm 1.9, the above algorithm can be shown to factor n with probability 1/2. This is a complete break of the cryptosystem.

5. Decribe attack models and adversarial goals for message authentication codes, by modifying the ones given for signature schemes in an appropriate manner.

   Answer: Suppose Alice and Bob are using a MAC with secret key K. The following attack models can be considered:

   known message attack
   The adversary possesses a list of messages and their authentication tags, say

   $$(x_1, y_1), (x_2, y_2), \ldots,$$

   where the x_i's are messages and the y_i's are the authentication tags on these messages (so $y_i = \mathsf{MAC}_K(x_i)$, $i = 1, 2, \ldots$).

   chosen message attack
   The adversary requests authentication tags for a list of messages. Therefore he chooses messages $x_1, x_2, \ldots$, and Alice or Bob supplies the authentication tags for these messages, namely, $y_i = \mathsf{MAC}_K(x_i)$, $i = 1, 2, \ldots$.

   Here is a list of possible adversarial goals:

   total break
   The adversary is able to determine the secret key K, i.e., the hash function MAC_K. Therefore he can create valid signatures on any message.

   selective forgery
   With some non-negligible probability, the adversary is able to create a valid authentication tags for a (random) message chosen by someone else. In other words, if the adversary is given a message x, then he can determine (with some probability) the authentication tag $y = \mathsf{MAC}_K(x)$. The message x should not be one for which the authentication tag has previously been determined in the relevant attack.

   existential forgery
   The adversary is able to create a valid authentication tag for at least one message. In other words, the adversary can create a pair (x, y) where x is a message and $\mathsf{MAC}_K(x) = y$. The message x should not be one for which the authentication tag has previously been determined in the relevant attack.

6. This question concerns MAC security.

Let $\ell \geq k \geq 1$ be integers. Suppose that a MAC is a keyed hash family $(\mathcal{X}, \mathcal{Y}, \mathcal{K}, \mathcal{H})$, where $\mathcal{X} = \{0,1\}^\ell$ and $\mathcal{Y} = \{0,1\}^k$ (thus a message is an ℓ-bit string and an authentication tag is a k-bit string). Suppose that the adversary has access to no previously authenticated messages and suppose that the message to be authenticated will be chosen at random. Finally, define

$$\epsilon = \max\left\{\frac{1}{|\mathcal{K}|}, \frac{1}{2^k}\right\}.$$

Give a complete proof that an adversary can compute the valid MAC for any message $x \in \mathcal{X}$ with probability at least ϵ.

Hint: Do *not* assume that the key K is chosen uniformly at random, and do *not* assume that all possible authentication tags are equally likely, given that the key is K.

Answer: Here is the algorithm, which tries to produce a valid authentication tag for a given message x.

(a) if $|\mathcal{K}| \leq 2^k$, then guess a key $K \in \mathcal{K}$ uniformly at random, and output $y = h_K(x)$.

(b) otherwise (if $|\mathcal{K}| > 2^k$) then guess a tag $y \in \mathcal{Y}$ uniformly at random, and output y.

Let K_0 be the "real" key that is being used by Alice and Bob, and let $y_0 = h_{K_0}(x)$. K_0 is chosen according to some probability distribution that is unknown to the adversary. Let $\mathbf{K_0}$ be the random variable taking on values in $\mathcal{K}$ according to Alice and Bob's probability distribution, and let $\mathbf{K}$ be the random variable taking on values in $\mathcal{K}$ according to the uniform probability distribution.

In (a), we claim that the following holds: $\mathsf{Pr}[\mathbf{K} = \mathbf{K_0}] = 1/|\mathcal{K}|$. This is easily proven as follows

$$\begin{aligned}
\mathsf{Pr}[\mathbf{K} = \mathbf{K_0}] &= \sum_{L \in \mathcal{K}} (\mathsf{Pr}[\mathbf{K} = L] \times \mathsf{Pr}[\mathbf{K_0} = L]) \\
&= \sum_{L \in \mathcal{K}} \left(\frac{1}{|\mathcal{K}|} \times \mathsf{Pr}[\mathbf{K_0} = L]\right) \\
&= \frac{1}{|\mathcal{K}|} \sum_{L \in \mathcal{K}} \mathsf{Pr}[\mathbf{K_0} = L] \\
&= \frac{1}{|\mathcal{K}|}.
\end{aligned}$$

Now, it is clear that $y = y_0$ whenever $K = K_0$, so the choice of y from (a) is correct with probablity at least $1/|\mathcal{K}|$.

Let $\mathbf{Y}$ be the random variable taking on values in $\mathcal{A}$ according to the uniform probability distribution. Let $\mathbf{Y_0}$ be the random variable taking on values in $\mathcal{A}$ as determined by Alice and Bob's probability distribution on $\mathcal{K}$. Then, in (b), we claim that $\mathsf{Pr}[\mathbf{Y} = \mathbf{Y_0}] = 1/2^k$. The proof is essentially identical to the previous proof.

We have proven that the adversary outputs a valid forgery in (a) with probability (at least) $1/|\mathcal{K}|$, and in (b) with probability $1/2^k$. The adversary will decide to do (a) or (b) based on which is more likely to succeed.