CS5430 Homework 2: Shared-key Cryptography

General Instructions. You are expected to work alone on this assignment.

Due: Sept 28, 2022 11:59pm. No late assignments will be accepted.

Submit your solution using CMS. Prepare your solution as .pdf, as follows:

- Use 10 point or larger font.
- Submit each problem (as a separate file) into the correct CMS submission box for that problem.

Assume that the threat is a Dolev-Yao attacker who is also able to perform type-attacks.

Problem 1: Machine Authentication. The following protocol is intended to allow a machine A to have assurance that it is communicating with a machine B. Machines A and B each are initially provisioned with a copy of shared key k_{AB}.

1. A: generate fresh shared key k_n each time the protocol will be run authenticating B, and store pair (B, k_n) for use later in this run of the protocol.
2. A → B: A , {A, k_n} k_{AB}
3. B → A: B, {A}k_n
4. A: Given "B" in field 1 of message 3, retrieve stored pair (B, k_x) Then use that information to check whether the following decryption holds:

$$k_x\text{-}D(\ \{A\}k_n\ \ ) = A?$$

Note, in message 2, the value in field 1("A" in the run given above) informs the receiver (B in the run given above) which key to use for decrypting field 2. So, receiver B would use either k_{AB} or k_{BA}. The encrypted "A" in field 2 of message 2 is used only as the input to the encryption function in message 3.

(a) Is it possible for an attacker T to use a reflection attack in order to impersonate B to A?

(b) Describe a simple modification to the protocol that would prevent your reflection attack.

Problem 2: Shared-key encryption with difference enc/dec keys. For bit strings b and c, the bitwise exclusive-or operation $\oplus$ is defined as follows for n-bit strings:

$$b \oplus c = d \quad \text{if and only if} \quad \text{for } 1 \le i \le n\text{:} \ d[i] = (b[i] + c[i]) \bmod 2$$

where x[i] indicates the i^{th} bit of bitstring x.

Given an n-bit message m, an n-bit secret encryption key ke, and an associated n-bit secret decryption key kd, we define the following encryption (E) and decryption (D) cryptographic operations. The notation k-F(x) is used to indicate the evaluation of a function F that has as its arguments an n-bit key k and an n-bit bitstring x. And the notation inverse(x) indicates the bit string obtaining by inverting all of the bits in x.

The definitions of E and D are the following:

ke-E(m): m $\oplus$ ke
kd-D(c): inverse(c $\oplus$ kd)

(a) Show that if ke = inverse(kd) holds then kd-D(ke-E(m)) = m holds for all m and, therefore, D reverses any obfuscation that E introduces.

(b) An encryption function implements *perfect secrecy* if and only if ciphertext reveals nothing about the plaintext. We can prove that an encryption function exhibits perfect secrecy by showing: for any ciphertext c (with length n) and any plaintext p (with length n) there exists some encryption key ke that, when used to encrypt p, produces c. That is, we are showing that any plaintext could be responsible for a given ciphertext (so the ciphertext is keeping the plaintext secret).

Show a function keygen(p,c) to generate the key for xor-Encryption of any plaintext p and ciphertext c. Show that your function suffices for proving perfect secrecy of k-E(p) or prove that no such function exists.

Problem 3: Analysis of an authentication protocol. The following key distribution protocol was inspired by Otway-Rees (which is described in the on-line lecture notes); the designer was concerned with the cost of encryption and therefore eliminated the encryptions used for Otway-Rees messages 1 and 2.

```
1. A   --> B:   n,A,B,r1    for fresh r1
2. B   --> KDC: n,A,B,r1,r2   for fresh r2
3. KDC --> B:   n,{r1,r2,A,B,K_AB}K_A, {r1,r2,A,B,K_AB}K_B
4. B   --> A:   {r1,r2,A,B,K_AB}K_A
```

We are interested in knowing what can be believed after the protocol terminates (assuming no participant has crashed). Below, we give an example (i) along with the answer, so you see the kind of analysis we are seeking.

i. What can `A` believe about `K_AB` and about which principals will know `K_AB` ?

   A can believe that only principals holding either K_A or K_B know K_AB. This set is at most {A,B,KDC}. A can also believe that K_AB is fresh since message 4 contains r1 which A created in message 1. Unlike Otway-Rees, A cannot assume that B participated in this run of the protocol (note that B does not encrypt any messages). Therefore the completion of the protocol cannot be used to determine that B interacted with A and therefore has the key.

ii. What can `B` believe about `K_AB` and about which principals will know `K_AB` ?

iii. What can `KDC` believe about `K_AB` and about which principals will know `K_AB?`