

CMSC 414: HW 5

1. Show a protocol for an augmented form of EKE.

Solution (one way to get EKE augmented):

A	B
----------	----------

A stores password pw

- W and W' are two keys obtained from pw (e.g., different hashes)

B stores (A, W', T_A') where $T_A' = g^W \text{ mod-p}$

Public g and p (prime)

- | | |
|--|--|
| <ul style="list-style-type: none"> ▪ choose m a ▪ $T_A \leftarrow g^a \text{ mod-p}$ ▪ send [A, B, W' {T_A}] | <ul style="list-style-type: none"> ▪ receive msg ▪ extract T_A from $W' \{T_A\}$ using W' ▪ choose m b ▪ $T_B \leftarrow g^b \text{ mod-p}$ ▪ $K_B \leftarrow (T_A)^b \text{ mod-p}$ ▪ $K_B' \leftarrow (T_A')^b \text{ mod-p}$ ▪ $H \leftarrow \text{hash}(K_B, K_B')$ ▪ send [B, A, W' {T_B}, H] |
| <ul style="list-style-type: none"> ▪ receive msg ▪ extract T_B from $W' \{T_B\}$ using W' ▪ $K_A \leftarrow (T_B)^a \text{ mod-p}$ ▪ $K_A' \leftarrow (T_B)^W \text{ mod-p}$ ▪ verify $H = \text{hash}(K_A, K_A')$ to authenticate B ▪ $H' \leftarrow \text{hash}'(K_A, K_A')$, where hash' is another hash function ▪ send [A, B, H'] | <ul style="list-style-type: none"> ▪ receive msg ▪ verify $H' = \text{hash}'(K_B, K_B')$ to authenticate A |

A and B are mutually authenticated and share strong key $K = g^{ab} \text{ mod p}$

2. (text 12.14) Why is the EKE-based Protocol 12-7 (also shown below) insecure? (Hint: someone impersonating Bob can do a dictionary attack, but show how.) How can you make it secure while still having Bob transmit $g^b \text{ mod-p}$ unencrypted?

A (pw, public g, p) // Alice	B (A: W, public g, p) // Bob
<ul style="list-style-type: none"> ▪ generate random a ▪ generate W from pw ▪ $T_A \leftarrow g^a \text{ mod-p}$ ▪ send [A, B, W{T_A}] ▪ receive msg ▪ $K_A \leftarrow (T_B)^a \text{ mod-p}$ ▪ generate challenge C_A ▪ send [A, B, K_A{C_B}, C_A] ▪ receive msg ▪ decrypt K_B{C_A} with K_A; if it equals C_A then B authenticated 	<ul style="list-style-type: none"> ▪ receive msg ▪ extract T_A from W{T_A} using W ▪ generate random b, challenge C_B ▪ $T_B \leftarrow g^b \text{ mod-p}$ ▪ $K_B \leftarrow (T_A)^b \text{ mod-p}$ ▪ send [B, A, T_B, C_B] ▪ receive msg ▪ decrypt K_A{C_B} with K_B; if it equals C_B then A authenticated ▪ send [B, A, K_B{C_A}]

Solution

Attacker: receive msg 1; generate b, C_B, T_B; send msg 2; receive msg 3.

Do following with W{T_A}, b, T_B, C_B, K{C_B} where $K = g^{ab} \text{ mod-p}$:

```

for cPw in dictionary do {
  cW ← key from cPw;
  cTA ← decrypt W{TA} using cW;
  cK ← (cTA)b mod-p;
  cCB ← decrypt K{CB} using cK;
  if cCB = CB then cPw equals pw
}
    
```

Fix: have B send evidence of K in msg 2, for example:

- B can send hash(K, C_B), or
- B can send K{C_B} instead of C_B, and the response can be K{C_B+1}.

3. (text 12.15) Consider Protocol 12-8. How would Alice compute K? How would Bob compute K? Why is it insecure? (Hint: someone impersonating Bob can do a dictionary attack, but show how.)

Solution

Bob computes K as follows:

- After receiving msg 1, Bob has $T_A' = g^W \text{ mod-p}$, $T_A = g^a \text{ mod-p}$, and b.
- $g^{ab} \text{ mod-p} = (T_A)^b \text{ mod-p}$
- $g^{Wb} \text{ mod-p} = (T_A')^b \text{ mod-p}$
- So $K \leftarrow \text{hash}((T_A)^b \text{ mod-p}, (T_A')^b \text{ mod-p})$

Alice computes K as follows:

- After receiving msg 2, Alice has $T_B = g^b \text{ mod-p}$, W, and a.
- $g^{ab} \text{ mod-p} = (T_B)^a \text{ mod-p}$
- $g^{Wb} \text{ mod-p} = (T_B)^W \text{ mod-p}$
- So $K \leftarrow \text{hash}((T_B)^a \text{ mod-p}, (T_B)^W \text{ mod-p})$

Dictionary attack by attacker impersonating Bob

Attacker: receive msg 1; generate b, C_1 ; send msg 2; receive msg 3.

So attacker has:

$T_A (= g^a \text{ mod-p})$; b; $T_B (= g^b \text{ mod-p})$; C_1 ; $K\{C_B\}$ where $K = g^{ab} \text{ mod-p}$

Do following:

```

for cPw in dictionary do {
  cW ← key from cPw;
  cGW ←  $g^{cW} \text{ mod-p}$ ;
  cK ←  $\text{hash}( (T_A)^b \text{ mod-p}, (cGW)^b \text{ mod-p} )$ ;
  cC1 ← decrypt  $K\{C_1\}$  using cK;
  if cC1 = C1 then cPw equals pw
}

```
