Key Management in Ad-Hoc Networks

Jukka Valkonen

Laboratory for Theoretical Computer Science
Helsinki University of Technology

28.3.2007

Outline

- Key management approaches
- Authenticated key agreement methods
- Case: WLAN

Key Management Approaches

1. Key predistribution
2. Key transport
3. Key arbitration
4. Key agreement

Now let's take a close look at these

Key Management Approaches (2)

- Key Predistribution
  - Key distributed to all aparties before communication
  - Static: Not possible to add devices
- Key Transport
  - One device generates a key, and transmits it to all receivers
  - The simplest scheme: Predistributed key is used to encrypt the session key. Also PKI is possible
  - Shamir's three-pass protocol (See next slide)

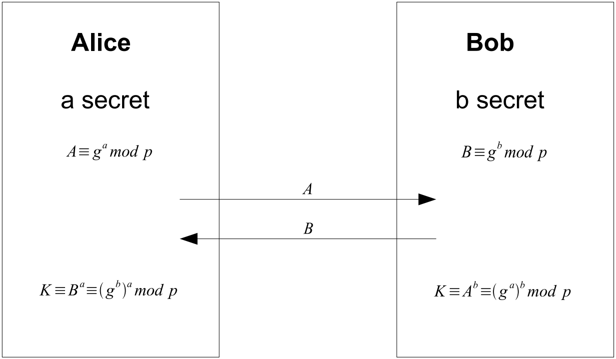
Shamir's three-pass protocol

1. D_1 generates random key K and encrypts it using f with random key x and sends the value to D_2
   $D_1 \longrightarrow D_2$: $f_x(K)$
2. D_2 encrypts the received message using g and a random key y and sends the value to D_1
   $D_1 \longleftarrow D_2$: $g_y(f_x(K))$
3. D_1 decrypts the received value using f^{-1} and x and sends the value to D_2
   $D_1 \longrightarrow D_2$:
   $f^{-1}(g_y(f_x(K))) = f_x^{-1}(f_x(g_y(K))) = g_y(K)$
4. D_2 decrypts the received value using g^{-1} and y.

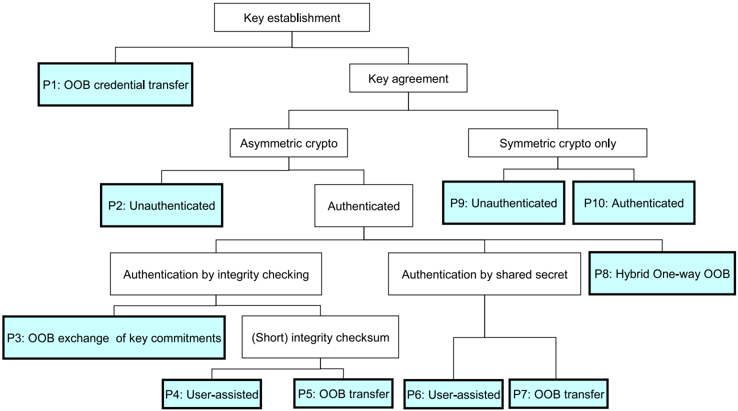
Key Management Approaches (3)

- Key Arbitration
  - Central arbitrator creates and distributes the keys
  - For example AP
  - The arbitrator need to be accessible by all the devices all the time
- Key Agreement
  - For example Diffie-Hellman Key agreement protocol
  - A passive attacker does not get the key, active man-in-the-middle is a threat
  - Needs quite a lot of computational power

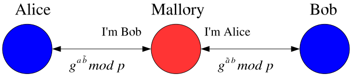
Recap: Diffie-Hellman Key Exchange



Another Classification



- Authentication is crucial aspect
- Diffie-Hellman key negotiation is vulnerable to active man-in-the-middle attacks



Encrypted Key Exchange

1. D_1 picks fresh random number a, and sends $(1, P(g^a \mod p))$ to D_2
2. D_2 picks fresh random number b, computes the shared secret $K = g^{ab} \mod p$, and sends $(P(g^b \mod p), K(challenge_2))$ to D_1
3. D_1 computes the shared secret $K = g^{ab} \mod p$, and sends $K(challenge_1, challenge_2)$ to D_2
4. D_2 verifies, that $challenge_2$ was echoed correctly and sends $K(challenge_1)$ to D_1.
5. D_1 verifies, that $challenge_1$ was echoed correctly

Encrypted Key Exchange for Groups

1. $D_i \longrightarrow D_{i+1} : g^{R_1 R_2 \dots R_i} \mod p$, $i = 1, \dots, n-2$
2. $D_{n-1} \longrightarrow$ ALL: $\pi = g^{R_1 R_2 \dots R_{n-1}} \mod p$
3. $D_i \longrightarrow D_n$: $P(c_i)$, $i = 1, \dots, n-1$, where $c_i = \pi^{\frac{\tilde{R}_i}{R_i}}$ and $\tilde{R}_i$ is a fresh random number generated by D_i
4. $D_n \longrightarrow D_i$: $c_i^{R_n}$, $i = 1, \dots, n-1$
5. $D_i \longrightarrow$ ALL: D_i, $K(D_i, H(D_1, D_2, \dots, D_n))$ for some i

Numeric Comparison: MANA IV

- Devices authenticate public Diffie-Hellman keys
- The users are expected to compare verification strings

1. $\mathrm{D}_1 \rightarrow \mathrm{D}_2$: $h(R_1)$
2. $\mathrm{D}_1 \leftarrow \mathrm{D}_2$: R_2
3. $\mathrm{D}_1 \rightarrow \mathrm{D}_2$: R_1
4. D_2 checks if $\hat{h} \stackrel{?}{=} h(\hat{R}_1)$ If equality holds, D_2 computes $v_2 = f(P\hat{K}_1, PK_2, \hat{R}_1, R_2)$, otherwise it aborts .

   D_1 computes $v_1 = f(PK_1, P\hat{K}_2, R_1, \hat{R}_2)$.
5. Both devices check if v_1 equals v_2.

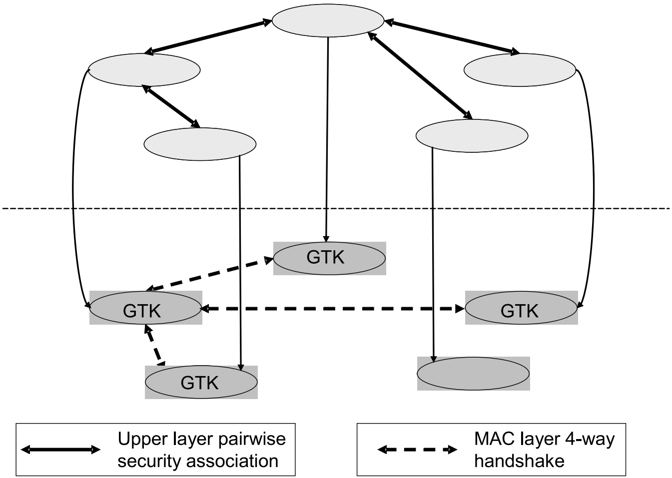
Threshold Cryptography

- Relies on trusted third parties
- $t + 1$ out of n servers needed to sign a certificate (where $n \geq 3t + 1$)
- Signing a certificate:
  - Each server generates a partial signature
  - The partial signatures are sent to a combiner
  - The combiners generates the signature out of $t + 1$ partial signatures and verifies it
  - If verification fails, at least one partial signature was not valid
  - New set of $t + 1$ partial signatures is tried

Multicast: WLAN

- Devices first negotiate upper layer keys (for example Simple Config)
- The upper layer keys are used on MAC-layer to negotiate keys
- A Group Temporal Key (GTK) is derived
    - Sender specific

Multicast: WLAN



Conclusions

- Key management is crucial aspect
- Multiple different ways to handle key negotiation
- Not enough to just negotiate key, but lower level protocols need to be taken into consideration also

Thank You!

Questions?