Introduction to Modern Cryptography

Lecture 12

Identification (User Authentication)

Zero Knowledge

Wrap Up

Lecture Outline

• Model
• Fiat-Shamir Identification Scheme (1987)
• Zero Knowledge (Goldwasser-Micali-Rackoff 1985)
• Wrap up

Model

• Alice wishes to prove to Bob her identity in order to access a resource, obtain a service etc.
• Bob may ask the following:
  – Who are you? (prove that you’re Alice)
  – Who the **** is Alice?
• Eve wishes to impersonate Alice:
  – One time impersonation
  – Full impersonation (identity theft)

Identification Scenarios

• Local identification
  – Human authenticator
  – Device
• Remote identification
  – Human authenticator
  – Corporate environment (e.g. LAN)
  – E-commerce environment
  – Cable TV/Satellite: Pay-per-view; subscription verification
  – Remote login or e-mail from an internet cafe.

Initial Authentication

• The problem: how does Alice initially convince anyone that she’s Alice?
• The solution must often involve a “real-world” type of authentication – id card, driver’s license etc.
• Errors due to the human factor are numerous (example – the Microsoft-Verisign fiasco).
• Even in scenarios where OK for Alice to be whoever she claims she is, may want to at least make sure Alice is human (implemented, e.g. for new users in Yahoo mail).

Closed Environments

• The initial authentication problem is fully solved by a trusted party, Carol
• Carol can distribute the identification material in a secure fashion, e.g. by hand, or over encrypted and authenticated lines
• Example – a corporate environment
• Eve’s attack avenue is the Alice-Bob connection
• We begin by looking at remote authentication
Fiat-Shamir Scheme

- Initialization
- Set Up
- Basic Construction
- Improved Construction
- Zero Knowledge
- Removing Interaction

Initialization

- Bob gets from Carol N=pq but not its factorization.
- Alice picks m numbers R1,R2,...,Rm in \( \mathbb{Z}_N \) at random.
- Alice computes \( S_1 = R_1^2 \mod N, \ldots, S_m = R_m^2 \mod N \).
- Alice gives Bob \( S_1,S_2,...,S_m \).
- She keeps \( R_1,R_2,...,R_m \) secret.

Set Up

- Bob holds \( S_1,S_2,...,S_m \).
- She keeps \( R_1,R_2,...,R_m \) secret.
- Who is Alice? Anyone that convinces Bob she can produce square roots mod N of \( S_1,S_2,...,S_m \).

- A bad way to convince Bob: Send him \( R_1,R_2,...,R_m \).
- Instead, we seek a method that will give Bob (and Eve) nothing more than being convinced Alice can produce these square roots (zero knowledge).

Basic Protocol

- Let \( S_1 = R_1^2 \) such that Alice holds \( R_1 \).
- To convince Bob that Alice knows a square root mod N of \( S_1 \), Alice picks at random \( X_1 \) in \( \mathbb{Z}_N \), computes \( Y_1 = X_1^2 \mod N \), and sends \( Y_1 \) to Bob.
- Alice: "I know both a square root mod N of \( Y_1 (=X_1) \) and a square root mod N of \( Y_1 S_1 (=X_1 R_1) \). Make a choice which of the two you want me to reveal."
- Bob flips a coin, outcome (heads/tails) determines the challenge he poses to Alice.

Basic Protocol (cont.)

- If Alice knows both a square root of \( Y_1 (=X_1) \) and a square root of \( Y_1 S_1 (=X_1 R_1) \) then she knows \( R_1 \) (a square root of \( S_1 \)).
- Thus if Alice does not know a square root of \( S_1 \), Bob will catch her cheating with probability 1/2.

- In the protocol, Alice will produce \( Y_1,Y_2,...,Y_m \).
- Bob will flip m coins \( b_1,b_2,...,b_m \) as challenges.
- Bob accepts only if Alice succeeds in all m cases.

Basic Protocol

Alice to Bob

\[ Y_1,Y_2,...,Y_m \]

Bob to Alice (challenge)

\[ b_1,b_2,...,b_m \]

1, 0, …, 0

Alice to Bob

\[ X_1,S_1,X_2,...,X_m \]

m responses

Bob accepts iff all m challenges are met.
Improved (more efficient) Protocol

Alice to Bob

\[ Y_1, Y_2, \ldots, Y_m \]

Bob to Alice (challenge)

\[ b_1, b_2, \ldots, b_m \]

Product of \( X_iR_i \) with \( b_i=1 \)

Product of \( X_i \) with \( b_i=0 \)

Alice to Bob (2 response)

Bob accepts iff challenges are met.

Correctness of Protocol (Intuition ONLY)

1. A cheating Eve, without knowledge of \( R_i \)'s, will be caught with high probability.

2. Zero Knowledge:
   By eavesdropping, Eve learns nothing (all she learns she can simulate on her own).

Crucial ingredients:
1. Interaction.
2. Randomness.

Final Improvement (Fiat Shamir)

Alice to Bob

\[ Y_1, Y_2, \ldots, Y_m \]

Bob to Alice (challenge)

\[ b_1b_2 \ldots b_m = H(Y_1, Y_2, \ldots, Y_m) \]

Product of \( X_iR_i \), \( b_i=1 \)

Product of \( X_i \), \( b_i=0 \)

Alice to Bob (2 response)

Bob accepts iff challenges are met.

And H be a secure hash function

Correctness of Fiat-Shamir (Intuition ONLY)

A cheating Eve, without knowledge of \( R_i \)'s, cannot succeed in producing \( Y_1, Y_2, \ldots, Y_m \) that will be hashed to a convenient bit vector \( b_1b_2 \ldots b_m \) since \( m \) is too long and \( H \) behaves like a random function (so the chances of hitting a bit vector favourable to Eve are negligible).

FS scheme used in practice.

Final Improvement: Remove Interaction

Alice to Bob

\[ Y_1, Y_2, \ldots, Y_m \]

Bob to Alice (challenge)

\[ b_1b_2 \ldots b_m = H(Y_1, Y_2, \ldots, Y_m) \]

Product of \( X_iR_i \), \( b_i=1 \)

Product of \( X_i \), \( b_i=0 \)

Alice to Bob (2 response)

Bob accepts iff challenges are met.

Let H be secure hash function

Course Outline (taken from Lecture 1)

- Encryption
- Data integrity
- Authentication and identification.
- Digital signatures.
- Number theory.
- Randomness and pseudo randomness.
- Cryptographic protocols.
- Real world security systems.