Cryptography
CS 555

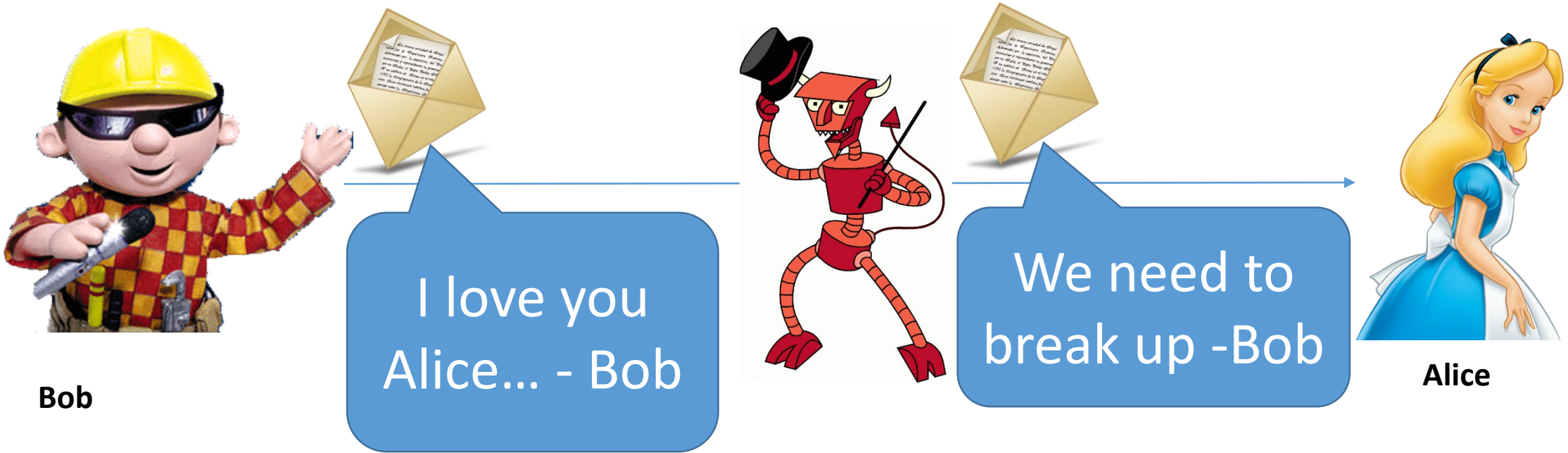
Topic 32: Digital Signatures Part 1

Recap

- CPA/CCA Security for Public Key Crypto
- Key Encapsulation Mechanism
- El-Gamal/RSA-OAEP

What Does It Mean to “Secure Information”

- Confidentiality (Security/Privacy)
  - Only intended recipient can see the communication
- Integrity (Authenticity)
  - The message was actually sent by the alleged sender



Encryption/MACs/Signatures

- (Public/Private Key) Encryption: Focus on Secrecy
  - But does not promise integrity
- MACs/Digital Signatures: Focus on Integrity
  - But does not promise secrecy
- Digital Signatures
  - Public key analogue of MAC

Digital Signature: Application

- Verify updates to software package
- Vendor generates (**sk**,**pk**) for Digital Signature scheme and packages **pk** in the original software bundle
- An update **m** should be signed by vendor using secret key **sk**
- **Security**: Malicious party should not be able to generate signature for new update **m'**

Digital Signature vs MACs

- Application: Validate updates to software
- Problem can be addressed by MACs, but there are several problems
- Key Explosion: Vendor must sign update using every individual key
  - Thought Question: Why not use a shared Private key?
- Non-Transferable: If Alice validates an update from vendor she can not convince Bob that the update is valid
  - Bob needs to receive MAC directly from vendor

Digital Signatures vs MACs

- Publicly Verifiable
- Transferable
  - Alice can forward digital signature to Bob, who is convinced (both Alice and Bob have the public key of the vendor)
- Non-repudiation
  - Can “certify” a particular message came from sender
- MACs do not satisfy non-repudiation
  - Suppose Alice reveals a shared key KAB along with a valid tag for a message m to a judge.
  - The judge should not be convinced the message was MACed by Bob. Why not?

Digital Signature Scheme

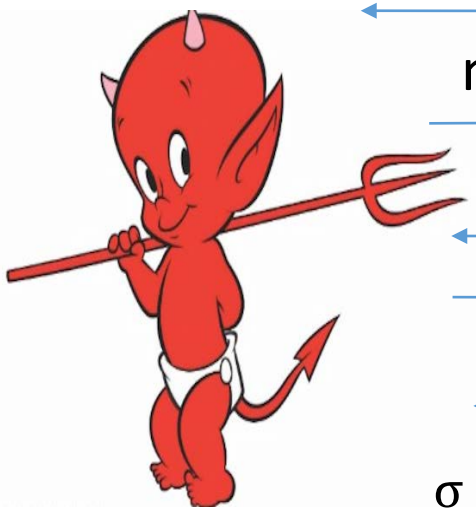
- Three Algorithms
  - $\mathrm{Gen}(1^n, R)$ (Key-generation algorithm)
    - Input: Random Bits R
    - Output: $(\boldsymbol{pk}, \boldsymbol{sk}) \in \mathcal{K}$
  - $\sigma \leftarrow \mathrm{Sign}_{\mathrm{sk}}(m, R)$ (Signing algorithm)
    - Input: Secret key sk message m, random bits R
    - Output: signature σ
  - $\mathrm{b} := \mathrm{Vrfy}_{\mathrm{pk}}(m, \sigma)$ (Verification algorithm --- Deterministic)
    - Input: Public key pk, message m and a signature σ
    - Output: 1 (Valid) or 0 (Invalid)

Alice must run key generation algorithm in advance an publishes the public key: pk

Assumption: Adversary only gets to see pk (not sk)

- **Correctness**: $\mathrm{Vrfy}_{\mathrm{pk}}(\mathrm{m}, \mathrm{Sign}_{\mathrm{sk}}(m, R))=1$ (except with negligible probability)

Signature Experiment $(\mathrm{Sig} - \mathrm{forge}_{\mathrm{A},\Pi}(\mathrm{n}))$



Public Key: pk

m_1

$\sigma_1 = \mathbf{Sign}_{sk}(\boldsymbol{m_1})$

m_2

$\sigma_2 = \mathbf{Sign}_{sk}(\boldsymbol{m_2})$

$\sigma, m \notin \mathfrak{Q} = \{m_1, m_2 \ldots\}$

$\mathrm{Sig} - \mathrm{forge}_{\mathrm{A},\Pi}(\mathrm{n}) = \mathrm{Vrfy}_{\mathrm{pk}}(m, \sigma)$

Random bit b

(pk,sk) = Gen(.)

$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$

$$\Pr\left[\mathrm{Sig} - \mathrm{forge}_{\mathrm{A},\Pi}(\mathrm{n}) = 1\right] \leq \mu(n)$$

Signature Experiment ($\mathrm{Sig} - \mathrm{forge}_{A,\Pi}(n)$)

Formally, let $\Pi = (\mathrm{Gen}, \mathrm{Sign}, \mathrm{Vrfy})$ denote the signature scheme, call the experiment $\mathrm{Sig} - \mathrm{forge}_{A,\Pi}(n)$

We say that Π *is existentially unforgeable under an adaptive chosen message attack* (or just *secure*) if for all PPT adversaries A, there *is a* negligible function μ such that

$$\Pr[\mathrm{Sig} - \mathrm{forge}_{A,\Pi}(n) = 1] \leq \mu(n)$$

Existential Unforgeability

- **Limitation**: Does not prevent replay attacks
  - $\sigma \leftarrow \mathrm{Sign}_{\mathrm{sk}}(\text{"}Pay\ Bob\ \$50\text{"}, R)$
  - If this is a problem then you can include timestamp in signature

- Does rule out the possibility of modifying a signature as in Homework 3
  - Homework 3: Plain RSA signatures are malleable

Hash and Sign Paradigm

- Public-Key vs Private Key Encryption
  - Private Key Encryption is much more efficient (computationally)
- Similarly, natural signature schemes (e.g., RSA signatures) are much less efficient than MACs

- For long messages we can achieve same (amortized) efficiency

Hash and Sign Paradigm

- Suppose we have a Digital Signature Scheme for messages of length $\ell(n)$ and we want to sign a longer message $m \in \{0,1\}^*$.
- **Attempt 1:**

$$\mathrm{Sign}^*_{\mathrm{sk}}\,(m_1, m_2, \ldots, m_k, R_1, \ldots, Rk) = \mathrm{Sign}^*_{\mathrm{sk}}(m_1, R_1), \ldots, \mathrm{Sign}^*_{\mathrm{sk}}(m_k, R_k)$$

- Problem?

Hash and Sign Paradigm

- Suppose we have a Digital Signature Scheme for messages of length $\ell(n)$ and we want to sign a longer message $m \in \{0,1\}^*$.

$$\mathrm{Sign}^*_{\langle sk,s\rangle}(m_1, m_2, \ldots, m_k, R) = \mathrm{Sign}^*_{\mathrm{sk}}(H(m_1, m_2, \ldots, m_k), R)$$

$$\mathrm{Vrfy}^*_{\langle sk,s\rangle}(m_1, m_2, \ldots, m_k, \sigma) = \mathrm{Vrfy}_{\mathrm{sk}}(H^s(m_1, m_2, \ldots, m_k), \sigma)$$

- Secure?

Hash and Sign Paradigm

- Suppose we have a Digital Signature Scheme for messages of length $\ell(n)$ and we want to sign a longer message $m \in \{0,1\}^*$.

$$\text{Sign}^*_{\langle sk,s\rangle}(m_1, m_2, \dots, m_k, R) = \text{Sign}^*_{\text{sk}}(H^s(m_1, m_2, \dots, m_k), R)$$

$$\text{Vrfy}^*_{\langle sk,s\rangle}(m_1, m_2, \dots, m_k, \sigma) = \text{Vrfy}_{\text{sk}}(H^s(m_1, m_2, \dots, m_k), \sigma)$$

- Secure?

Theorem 12.4. If $\Pi = (\text{Gen}, \text{Sign}, \text{Vrfy})$ is a secure signature scheme for messages of length $\ell(n)$ and Π_H is collision resistant then the above construction is a secure signature scheme for arbitrary length messages.

Hash and Sign Paradigm

- Suppose we have a Digital Signature Scheme for messages of length $\ell(n)$ and we want to sign a longer message $m \in \{0,1\}^*$.

$$\mathrm{Sign}^*_{\langle sk,s\rangle}(m_1, m_2, \ldots, m_k, R) = \mathrm{Sign}_{\mathrm{sk}}(H^s(m_1, m_2, \ldots, m_k), R)$$

$$\mathrm{Vrfy}^*_{\langle sk,s\rangle}(m_1, m_2, \ldots, m_k, \sigma) = \mathrm{Vrfy}_{\mathrm{sk}}(H^s(m_1, m_2, \ldots, m_k), \sigma)$$

Theorem 12.4. If $\Pi = (\mathrm{Gen}, \mathrm{Sign}, \mathrm{Vrfy})$ is a secure signature scheme for messages of length $\ell(n)$ and Π_H is collision resistant then the above construction is a secure signature scheme for arbitrary length messages.

Proof Sketch: If attacker wins security game with $\mathrm{Sign}^*_{\langle sk,s\rangle}$ then he outputs message $m \notin \mathcal{Q}$ such that $\mathrm{Vrfy}^*_{\langle pk,s\rangle}(m, \sigma)$

Hash and Sign Paradigm

- Suppose we have a Digital Signature Scheme for messages of length $\ell(n)$ and we want to sign a longer message $m \in \{0,1\}^*$.

$$\text{Sign}^*_{\langle sk,s\rangle}(m_1, m_2, \dots, m_k, R) = \text{Sign}_{\text{sk}}(H^s(m_1, m_2, \dots, m_k), R)$$
$$\text{Vrfy}^*_{\langle sk,s\rangle}(m_1, m_2, \dots, m_k, \sigma) = \text{Vrfy}_{\text{sk}}(H^s(m_1, m_2, \dots, m_k), \sigma)$$

Theorem 12.4. If $\Pi = (\text{Gen}, \text{Sign}, \text{Vrfy})$ is a secure signature scheme for messages of length $\ell(n)$ and Π_H is collision resistant then the above construction is a secure signature scheme for arbitrary length messages.

Proof Sketch: If attacker wins security game with $\text{Sign}^*_{\langle sk,s\rangle}$ then he outputs message $m \notin \mathcal{Q}$ such that $\text{Vrfy}^*_{\langle pk,s\rangle}(m, \sigma)$

- Case 1: H(m)=H(m') for some $m' \notin \mathcal{Q}$

→break collision-resistance

- Case 2: H(m)≠ H(m') for all $m' \notin \mathcal{Q}$

→(break security of underlying signature scheme Π)

One-Time Signature Scheme

- Weak notion of one-time secure signature schemes
  - Attacker makes one query to oracle $Sign_{sk}(.)$ and then attempts to output forged signature for m’
  - If attacker sees two different signatures then guarantees break down
- Achievable from Hash Functions
  - No number theory!
  - No Random Oracles!

Lamport's Signature Scheme

$$sk = \begin{bmatrix} x_{1,0} & x_{2,0} & x_{3,0} \\ x_{1,1} & x_{2,1} & x_{3,1} \end{bmatrix}$$

$$pk = \begin{bmatrix} y_{1,0} & y_{2,0} & y_{3,0} \\ y_{1,1} & y_{2,1} & y_{3,1} \end{bmatrix}$$

$$x_{i,j} \in \{0,1\}^n \ (uniform)$$

$$y_{i,j} = H(x_{i,j})$$

Lamport's Signature Scheme

$$sk = \begin{bmatrix} x_{1,0} & x_{2,0} & x_{3,0} \\ x_{1,1} & x_{2,1} & x_{3,1} \end{bmatrix}$$

$$pk = \begin{bmatrix} y_{1,0} & y_{2,0} & y_{3,0} \\ y_{1,1} & y_{2,1} & y_{3,1} \end{bmatrix}$$

$$Sign_{sk}(011) = (x_{1,0}, x_{2,1}, x_{3,1})$$

Lamport's Signature Scheme

$$sk = \begin{bmatrix} x_{1,0} & x_{2,0} & x_{3,0} \\ x_{1,1} & x_{2,1} & x_{3,1} \end{bmatrix}$$

$$pk = \begin{bmatrix} y_{1,0} & y_{2,0} & y_{3,0} \\ y_{1,1} & y_{2,1} & y_{3,1} \end{bmatrix}$$

$$Sign_{sk}(011) = (x_{1,0}, x_{2,1}, x_{3,1})$$

$$\mathrm{Vrfy}_{pk}(011, (x_1, x_2, x_3)) = \begin{cases} 1 & \text{if } H^S(x_1) = y_{1,0} \wedge H^S(x_2) = y_{2,1} \wedge H^S(x_3) = y_{3,1} \\ 0 & \text{otherwise} \end{cases}$$

Lamport's Signature Scheme

Theorem 12.16: Lamport's Signature Scheme is a secure one-time signature scheme (assuming H is a one-way function).

Proof Sketch: Signing a fresh message requires inverting $H(x_{i,j})$ for some fresh i,j.

Remark: Attacker can break scheme if he can request two signatures.

How?

Request signatures of both 0^n and 1^n.

Lamport's Signature Scheme

Remark: Attacker can break scheme if he can request two signatures.

How?

Request signatures of both 0^n and 1^n.

$$sk = \begin{bmatrix} x_{1,0} & x_{2,0} & x_{3,0} \\ x_{1,1} & x_{2,1} & x_{3,1} \end{bmatrix}$$

$$Sign_{sk}(000) = (x_{1,0}, x_{2,0}, x_{3,0})$$

$$Sign_{sk}(111) = (x_{1,1}, x_{2,1}, x_{3,1})$$

Secure Signature Scheme from OWFs

Remark: Possible to construct signature scheme Π which is existentially unforgeable under an adaptive chosen message attacks using the minimal assumption that one-way functions exist.

Theorem 12.22: secure/stateless signature scheme from collision-resistant hash functions.

- Collision Resistant Hash Functions do imply OWFs exist

Next Class: Digital Signatures Part 2

- Read Katz and Lindell: 12.4-12.5