B504/I538: Introduction to Cryptography

Spring 2017 • Lecture 24
(2017-04-06)
RSA + OAEP

- OAEP = “Optimal Asymmetric Encryption Padding”
  - $n + k_1 + k_2$ is bit length of $N = pq$
  - $k_1$ and $k_2$ are constants (e.g., $k_1 = k_2 = 128$)
  - $n$ is plaintext length
  - $G$ and $H$ are “random oracles” (cryptographic hash functions)
    - $G: \{0, 1\}^{k_2} \rightarrow \{0, 1\}^{n + k_1}$
    - $H: \{0, 1\}^{n + k_1} \rightarrow \{0, 1\}^{k_2}$
  - $\oplus$ is bitwise XOR
- Public key is $(N, e, G, H)$
- Private key is $d := e^{-1} \mod \phi(N)$
RSA + OAEP

Encrypting $m \in \{0, 1\}^n$:

1. Pad $m$ with $k_1$ zeros: $m' := m \| 0^{k_1}$
2. Choose $r \in \{0, 1\}^{k_2}$
3. "Expand" $r$ using $G$: $r' := G(r)$
4. XOR $m'$ with $r'$: $X := m' \oplus r'$
5. "Compress" $X$ using $H$: $X' := H(X)$
6. XOR $X'$ with $r$: $Y := r \oplus X'$
7. Concatenate $X$ and $Y$: $m'' := X \| Y$
8. Apply "textbook RSA" to $m$": $c := (m'')^e$

Ciphertext is $c \in \mathbb{Z}_N$. 
RSA + OAEP

Decrypting $c \in \mathbb{Z}_N$

1. Recover $m''$ from $c$: $m'' := c^d \text{ mod } N$
2. Parse $m''$ as $(X, Y) \in \{0,1\}^{n+k_1} \times \{0,1\}^{k_2}$
3. Recover $X'$ from $X$: $X' := H(X)$
4. Recover $r$ from $X'$ and $Y$: $r := Y \oplus H(X)$
5. "Expand" $r$ using $G$: $r' := G(r)$
6. Recover $m'$ from $X$ and $r'$: $m' := X \oplus r'$
7. Parse $m'$ as $(m, z) \in \{0,1\}^n \times \{0,1\}^{k_1}$
8. Check $z \neq 0^{k_1}$; if not, abort

Plaintext is $m \in \{0,1\}^n$. 

$0^{k_1}$
Going forward

Tuesday (04/06): Digital signature schemes
Thursday (04/11): Random oracle model
Tuesday (04/13): Zero-knowledge proofs
Thursday (04/18): ??
Tuesday (04/20): ??
Thursday (12/25): ??
Thursday (04/27): ??

some possible topics
- Specific cryptosystems
- Secret sharing schemes
- Pairing-based cryptography
- Zero-Knowledge proofs
- Private information retrieval
- Secure multiparty computation
- Side-channel attacks
- Other security notions
- Anonymous credentials
- Lattice crypto
Recall: Secrecy versus Authenticity

- **Secrecy / confidentiality**
  - IND-CPA: indistinguishable multiple encryptions in the presence of an eavesdropper
  - Provides "security" in the presence of **passive attackers**
    - "security" == secrecy

- **Authenticity / integrity**
  - Existential unforgeability under adaptive chosen message attacks
  - Provides "security" in the presence of **active attackers**
    - "security" == integrity
Recall: MAC schemes

Defn: A message authentication code (MAC) is a triple of efficient algorithms (Gen, MAC, Ver) such that
- Gen: $\mathbb{N} \rightarrow K$ is a randomized “key generation” algorithm
- MAC: $K \times M \rightarrow T$ is a “tagging” algorithm
- Ver: $K \times M \times T \rightarrow \{0, 1\}$ is a “tag verification” algorithm

Usually write $\text{MAC}_k(m)$ and $\text{Ver}_k(m, t)$ instead of $\text{MAC}(k, m)$ and $\text{Ver}(k, m, t)$

- $K$ is the key space (the set of possible keys)
- $M$ is the message space (the set of possible messages)
- $T$ is the tag space (the set of possible “tags”)
Recall: MAC existential forgery game

Challenger (C)

1\textsuperscript{s} \leftarrow \text{Gen}(1\textsuperscript{s})

k \leftarrow \text{Gen}(1\textsuperscript{s})

m_1 \leftarrow \text{MAC}_k(m_1)

m_1 \in M

t_1 \leftarrow \text{MAC}_k(m_1)

t_1 \in T

m_2 \leftarrow \text{MAC}_k(m_2)

m_2 \in M

t_2 \leftarrow \text{MAC}_k(m_2)

t_2 \in T

\ldots

m_n \leftarrow \text{MAC}_k(m_n)

m_n \in M

t_n \leftarrow \text{MAC}_k(m_n)

t_n \in T

Attacker (A)

\text{Ver}_k(m, t) = 1

(m, t) \in M \times T

Let \( E \) be the event that \((m, t) \notin \{(m_1, t_1), \ldots, (m_n, t_n)\} \) yet \( \text{Ver}_k(m, t) = 1 \)

Define A's advantage to be \( \text{Adv}^{\text{MAC-forgery}}(A) = \text{Pr}[E] \)
Recall: Existential unforgeability

**Defn:** A MAC scheme \((\text{Gen}, \text{MAC}, \text{Ver})\) is existentially unforgeable under adaptive chosen message attacks if, for every PPT attacker \(A\), there exists a negligible function \(\varepsilon : \mathbb{N} \rightarrow \mathbb{R}^+\) such that

\[
\text{Adv}^{\text{MAC-forg}}(A) \leq \varepsilon(s).
\]

The message \(m\) and tag \(t\) are chosen arbitrarily by the attacker at the end of the attack.

Existential unforgeability is the "default" unforgeability property.
Digital signature schemes

- Intuitively, a digital signature scheme is the public-key equivalent of a MAC scheme.

**Defn:** A digital signature scheme is a triple of efficient algorithms \((\text{Gen}, \text{Sign}, \text{Ver})\) such that

- \(\text{Gen} : \mathbb{N} \to K_s \times K_v\) is a randomized “key generation” algorithm
- \(\text{Sign} : K_s \times M \to S\) is a randomized “signing” algorithm
- \(\text{Ver} : K_v \times S \to \{0, 1\}\) is a deterministic “signature verification” algorithm
Correctness

- Intuitively: Correctness is the property of being able to verify a signature (given the appropriate public key).

**Defn:** A signature scheme \((Gen, Sign, Ver)\) with message space \(M\) is correct if there exists a negligible function \(\varepsilon: \mathbb{N} \rightarrow \mathbb{R}^+\) such that, \(\forall s \in \mathbb{N}\) and \(\forall m \in M\),

\[
\Pr[Ver_{k_v}(m, Sign_{k_s}(m)) = 1 \mid (k_s, k_v) \leftarrow Gen(1^s)] \geq 1 - \varepsilon(s)
\]

If \(Ver(k_v, m, \sigma) \neq 1\), then we call \(\sigma\) a valid signature on \(m\) (under the verification key \(k_v\)).
Signature existential forgery game

\[ \text{Challenger (C)} \]

\[ (k_s, k_v) \leftarrow \text{Gen}(1^s) \]

\[ \sigma_1 \leftarrow \text{Sign}_{k_s}(m_1) \]

\[ \sigma_2 \leftarrow \text{Sign}_{k_s}(m_2) \]

\[ \sigma_n \leftarrow \text{Sign}_{k_s}(m_n) \]

\[ \text{Attacker (A)} \]

\[ m_1 \in M \]

\[ \sigma_1 \leftarrow \text{Sign}_{k_s}(m_1) \]

\[ \sigma_2 \leftarrow \text{Sign}_{k_s}(m_2) \]

\[ \sigma_n \leftarrow \text{Sign}_{k_s}(m_n) \]

\[ m_2 \in M \]

\[ m_n \in M \]

Let \( E \) be the event that \( (m, \sigma) \notin \{(m_1, \sigma_1), \ldots, (m_n, \sigma_n)\} \) yet \( \text{Ver}_{k_v}(m, \sigma) = 1 \)

Define A's advantage to be \( \text{Adv}_{\text{Sig-forgery}}(A) = \Pr[E] \)
Existential unforgeability

- Intuitively: Existential unforgeability is the property of being resistant to forgeries, even those arising from malicious tampering with existing signatures.

Defn: A digital signature scheme $(Gen, Sign, Ver)$ is existentially unforgeable under adaptive chosen message attacks if, for every PPT attacker $A$, there exists a negligible function $\varepsilon: \mathbb{N} \to \mathbb{R}^+$ such that

$$\text{Adv}_{\text{Sig-forg}}(A) \leq \varepsilon(s).$$
"Textbook" RSA signatures

- Many textbooks (and courses) describe RSA signatures as a direct application of the RSA trapdoor permutation to a message; that is,
  - \( \text{Gen}(1^s) \) chooses distinct \( s \)-bit primes \( p \) and \( q \) and \( e \in \mathbb{Z}_\varphi(pq) \), and then it outputs \( k_v := (pq, e) \) and \( k_s := e^{-1} \mod \varphi(pq) \);
  - \( \text{Sign}(k_s, m) \) outputs \( \sigma := m^d \mod pq \); and
  - \( \text{Ver}(k_v, m, \sigma) \) outputs 1 if \( m^e \equiv \sigma^e \mod pq \) and 0 otherwise.

Q: Is this an existentially unforgeable signature scheme?
A: NO! NO! NO! Don't ever do this! (Seriously, don't do it!)

(If you do this and I find out, I will retroactive fail you in this course!)
"Textbook" RSA signatures

As with textbook RSA encryption, textbook RSA signatures have some serious problems!

- Unclear how to sign "long" messages
- Extremely inefficient
- "No message" forgery attacks
- Forgeries from malleability
- Forgeries on arbitrary messages
- Key reuse attacks
Obs 1: Let $p_v := (pq, e)$ be a verification key for the textbook RSA signature scheme. Given any $\sigma \in \mathbb{Z}_N^*$, the attacker can output a valid forgery $(m, \sigma)$ where $m := \sigma^e \mod pq$.

Q: Can attacker choose $\sigma$ corresponding to a particular message of its choosing?
A: No! (Unless the RSA assumption fails to hold.)

- Message is an $e$th root of $\sigma$ modulo $pq$. 

Forgeries from malleability

**Obs 2**: Let $p_v := (pq, e)$ be a verification key for the textbook RSA signature scheme.

Given any two message-signatures pairs $(m_1, \sigma_1), (m_2, \sigma_2)$ such that $\text{Ver}(p_v, m_1, \sigma_1) = \text{Ver}(p_v, m_2, \sigma_2) = 1$, the attacker can output a valid forgery $(m', \sigma')$ where $m' := m_1^a m_2^b \mod pq$ and $\sigma' := \sigma_1^a \sigma_2^b \mod pq$.

- **Attacker can choose** $a$ and $b$ arbitrarily
  - $a = b = 1$ yields signature on $m_1 m_2$
  - $b = 0$ yields signature on $m_1^a$
Obs 3: Let $p_v = (pq, e)$ be a verification key for the textbook RSA signature scheme. Attacker can forge signature on arbitrary message $m$ of its choosing by

1. choosing any $r \in \mathbb{Z}_{pq}^*$ such that $r \neq 1$,
2. requesting a signature $\sigma'$ on $m' := m \cdot r$,
3. outputting $\sigma := \sigma' r^{-1} \mod pq$.

This “attack” is useful for constructing blind RSA signatures.
One-time signature schemes

- Intuitively, a one-time signature (OTS) scheme is a digital signature scheme that is existentially unforgeable provided the private key is only used to sign a single message.

Q: Why is this notion useful?

A1: We can construct OTS schemes under weaker assumptions
   - No random oracles, no computational assumptions

A2: OTS’s are a useful building block for “many-time” signatures
One-time forgery game

Let $E$ be the event that $(m', \sigma') \neq (m, \sigma)$ yet $\text{Ver}_{k_v}(m', \sigma') = 1$

Define $A$'s advantage to be $\text{Adv}_{\text{OTS-forgery}}(A) = \Pr[E]$
One-time existential unforgeability

Defn: A digital signature scheme $(\text{Gen}, \text{Sign}, \text{Ver})$ is existentially unforgeable under single-message attacks (or is one-time secure) if, for every PPT attacker $A$, there exists a negligible function $\varepsilon: \mathbb{N} \to \mathbb{R}^+$ such that

$$\text{Adv}^\text{OTS\text{-forge}}(A) \leq \varepsilon(s).$$
Lamport's OTS scheme (for $\ell(s)$-bit messages)

- Let $F$ be an OWF (In practice, $F$ is a cryptographic hash function).
  - $\text{Gen}(1^s)$ chooses $(x_{1,0}, x_{1,1}), \ldots, (x_{\ell(s),0}, x_{\ell(s),1}) \in \{0,1\}^s \times \{0,1\}^s$ and outputs
    
    $K_s := \{x_{1,0}, \ldots, x_{\ell(s),0}\}$ and $K_v := \{y_{1,0}, \ldots, y_{\ell(s),0}\}$, where each $y_{ij} := F(x_{ij})$.

  - $\text{Sign}(k_s, m_1 || m_2 || \cdots || m_{\ell(s)})$ outputs $\sigma := (x_{1,m_1}, x_{2,m_2}, \ldots, x_{\ell(s),m_{\ell(s)}})$.

  - $\text{Ver}(k_v, m_1 || m_2 || \cdots || m_{\ell(s)}, \sigma)$ outputs 1 if $y_{i,m_i} = F(x_{i,m_i})$ for each $i = 1, \ldots, \ell(s)$ and outputs 0 otherwise.
Thm: Let $\ell: \mathbb{N} \to \mathbb{N}$ be any positive integer-valued polynomial and let $F$ be an OWF. Then Lamport's OTS scheme for messages of length $\ell(s)$ is existentially unforgeable under single-message attacks.

Proof (Sketch): Suppose $A$ requests a signature on an $\ell(s)$-bit message $m$ and consider any message $m' \in \{0,1\}^s \setminus \{m\}$. Since $m' \neq m$, there exists a bit position $i$ for which $m_i \neq m'_i$; thus, forging a signature on $m'$ requires $A$ to find $x_i \in F^{-1}(y_{i,m})$. Since $F$ is an OWF, $A$ succeeds in doing so with negligible advantage. ☐
That's all for today, folks!