The Setting

- Alice and Bob don’t share any secret
- Alice wants to send a private message $m$ to Bob
- Goals:
  - **Public key:** Encryption and decryption keys are different. Encryption key can be “public”
  - **Correctness:** Alice can compute an encryption $c$ of $m$ using $pk$. Bob can decrypt $m$ from $c$ correctly using $sk$
  - **Security:** No eavesdropper can distinguish between encryptions of $m$ and $m'$ (even using $pk$)
Definition

- **Syntax:**
  - Gen$(1^n) \rightarrow (pk, sk)$
  - Enc$(pk, m) \rightarrow c$
  - Dec$(sk, c) \rightarrow m'$ or ⊥

  All algorithms are polynomial time

- **Correctness:** For every $m$, Dec$(sk, Enc(pk, m)) = m$, where $(pk, sk) \leftarrow \text{Gen}(1^n)$

- **Security:** ?
Definition ((Weak) Indistinguishability Security)

A public-key encryption scheme \((\text{Gen, Enc, Dec})\) is weakly indistinguishably secure under chosen plaintext attack (weak IND-CPA) if for all n.u. PPT adversaries \(A\), there exists a negligible function \(\mu(\cdot)\) s.t.:

\[
\Pr \left[ \begin{array}{c}
(pk, sk) \xleftarrow{\$} \text{Gen}(1^n), \\
(m_0, m_1) \xleftarrow{\$} A(1^n), \quad : A(pk, \text{Enc}(pk, m_b)) = b
\end{array} \right] \leq \frac{1}{2} + \mu(n)
\]

\(b \xleftarrow{\$} \{0, 1\}\)

1. Think: Semantic security style definition?
2. Think Equivalence of above definition and semantic security
A stronger definition:

**Definition (Indistinguishability Security)**

A public-key encryption scheme \((\text{Gen}, \text{Enc}, \text{Dec})\) is indistinguishably secure under chosen plaintext attack (IND-CPA) if for all n.u. PPT adversaries \(A\), there exists a negligible function \(\mu(\cdot)\) s.t.:

\[
\Pr \left[ \begin{array}{c}
(pk, sk) \leftarrow \text{Gen}(1^n), \\
(m_0, m_1) \leftarrow A(1^n, pk), \quad : A(pk, \text{Enc}(m_b)) = b \\
b \leftarrow \{0, 1\}
\end{array} \right] \leq \frac{1}{2} + \mu(n)
\]

1. Think: IND-CPA is stronger than weak IND-CPA
2. Think Multi-message security?
Lemma (Multi-message security)

*One-message security implies multi-message security for public-key encryption*
Multi-message security

**Lemma (Multi-message security)**

*One-message security implies multi-message security for public-key encryption*

1. **Think**: Proof?
Multi-message security

Lemma (Multi-message security)

One-message security implies multi-message security for public-key encryption

1. Think: Proof?
2. Corollary: Suffices to consider single-bit message
A collection of one-way functions is a family $\mathcal{F} = \{f_i : D_i \rightarrow R_i\}_{i \in \mathcal{I}}$ satisfying the following conditions:

1. **Sampling function:** There exists a PPT algorithm $\text{Gen}$ such that $\text{Gen}$ outputs an element uniformly random from $D_i$.

2. **Sampling from domain:** There exists a PPT algorithm that on input $i$ outputs a uniformly random element of $D_i$.

3. **Evaluation:** There exists a PPT algorithm that on input $i, x \in D_i$ outputs $f_i(x) = y$.

4. **Hard to invert:** For every PPT adversary $A$, there exists a negligible function $\mu$ such that:

   $$\Pr_{r \leftarrow \text{Gen}, y \leftarrow f_i(x)}[A(r, i, y) = 1] < \mu \cdot (\lambda)$$
One-way Functions, Revisited

Definition (Collection of OWFs)

A collection of one-way functions is a family \( \mathcal{F} = \{ f_i : D_i \rightarrow R_i \} \) satisfying the following conditions:

- **Sampling function:** There exists a PPT \( \text{Gen} \) s.t. \( \text{Gen}(1^n) \) outputs \( i \in \mathcal{I} \)

- **Evaluation:** There exists a PPT algorithm that on input \( i, x \) outputs \( f_i(x) \)

- **Hard to invert:** For every n.u. PPT adversary \( A \), there exists a negligible function \( \mu \) s.t.:
  \[
  \Pr_{r,i \leftarrow \text{Gen}(1^n), x \leftarrow D_i, y \leftarrow f_i(x)}[A(r,i,y) \neq x] < \mu(n)
  \]
One-way Functions, Revisited

Definition (Collection of OWFs)

A collection of one-way functions is a family $\mathcal{F} = \{f_i : D_i \rightarrow R_i\}_{i \in \mathcal{I}}$ satisfying the following conditions:

- **Sampling function:** There exists a PPT $\text{Gen}$ s.t. $\text{Gen}(1^n)$ outputs $i \in \mathcal{I}$

- **Sampling from domain:** There exists a PPT algorithm that on input $i$ outputs a uniformly random element of $D_i$
One-way Functions, Revisited

Definition (Collection of OWFs)

A collection of one-way functions is a family $\mathcal{F} = \{f_i : \mathcal{D}_i \to \mathcal{R}_i\}_{i \in \mathcal{I}}$ satisfying the following conditions:

- **Sampling function:** There exists a PPT $\text{Gen}$ s.t. $\text{Gen}(1^n)$ outputs $i \in \mathcal{I}$

- **Sampling from domain:** There exists a PPT algorithm that on input $i$ outputs a uniformly random element of $\mathcal{D}_i$

- **Evaluation:** There exists a PPT algorithm that on input $i, x \in \mathcal{D}_i$ outputs $f_i(x)$

- **Hard to invert:** For every n.u. PPT adversary $A$, there exists a negligible function $\mu$ s.t.:
  \[
  \Pr[r \leftarrow \text{Gen}(1^n), x \leftarrow \mathcal{D}_i, y \leftarrow f_i(x); A(r, i, y) = 1] \leq \mu(n)
  \]
One-way Functions, Revisited

Definition (Collection of OWFs)

A collection of one-way functions is a family \( \mathcal{F} = \{ f_i : D_i \rightarrow R_i \}_{i \in \mathcal{I}} \) satisfying the following conditions:

- **Sampling function:** There exists a PPT \( \text{Gen} \) s.t. \( \text{Gen}(1^n) \) outputs \( i \in \mathcal{I} \)

- **Sampling from domain:** There exists a PPT algorithm that on input \( i \) outputs a uniformly random element of \( D_i \)

- **Evaluation:** There exists a PPT algorithm that on input \( i, x \in D_i \) outputs \( f_i(x) \)

- **Hard to invert:** For every n.u. PPT adversary \( A \), there exists a negligible function \( \mu(\cdot) \) s.t.:

\[
\Pr \left[ i \leftarrow \text{Gen}(1^n) , x \leftarrow D_i, y \leftarrow f_i(x) : f_i (A (1^n, i, y)) = y \right] \leq \mu(n)
\]
Theorem

There exists a collection of one-way functions iff there exists a strong one-way function

Think: Proof?
A collection $\mathcal{F} = \{f_i : D_i \rightarrow R_i\}_{i \in \mathcal{I}}$ is a collection of one-way permutations if $\mathcal{F}$ is a collection of OWFs and for every $i \in \mathcal{I}$, $f_i$ is a permutation.
Definition (Trapdoor OWPs)

A collection of trapdoor permutations is a family of permutations \( \mathcal{F} = \{ f_i : D_i \rightarrow R_i \}_{i \in I} \) satisfying the following properties:
Trapdoor Permutations

**Definition (Trapdoor OWPs)**

A collection of trapdoor permutations is a family of permutations $\mathcal{F} = \{f_i : \mathcal{D}_i \rightarrow \mathcal{R}_i\}_{i \in \mathcal{I}}$ satisfying the following properties:

- **Sampling function:** $\exists$ a PPT Gen s.t. $\text{Gen}(1^n)$ outputs $(i, t) \in \mathcal{I}$
Trapdoor Permutations

Definition (Trapdoor OWPs)
A collection of trapdoor permutations is a family of permutations \( F = \{ f_i : D_i \rightarrow R_i \}_{i \in \mathcal{I}} \) satisfying the following properties:

- **Sampling function:** \( \exists \) a PPT Gen s.t. \( \text{Gen}(1^n) \) outputs \((i, t) \in \mathcal{I}\)
- **Sampling from domain:** \( \exists \) a PPT algorithm that on input \( i \) outputs a uniformly random element of \( D_i \)
Definition (Trapdoor OWPs)

A collection of trapdoor permutations is a family of permutations \( F = \{ f_i : D_i \rightarrow R_i \}_{i \in \mathcal{I}} \) satisfying the following properties:

- **Sampling function:** \( \exists \) a PPT Gen s.t. \( \text{Gen}(1^n) \) outputs \( (i, t) \in \mathcal{I} \)

- **Sampling from domain:** \( \exists \) a PPT algorithm that on input \( i \) outputs a uniformly random element of \( D_i \)

- **Evaluation:** \( \exists \) PPT that on input \( i, x \in D_i \) outputs \( f_i(x) \)
A collection of trapdoor permutations is a family of permutations \( \mathcal{F} = \{ f_i : \mathcal{D}_i \to \mathcal{R}_i \}_{i \in \mathcal{I}} \) satisfying the following properties:

- **Sampling function:** \( \exists \) a PPT Gen s.t. \( \text{Gen}(1^n) \) outputs \( (i, t) \in \mathcal{I} \)
- **Sampling from domain:** \( \exists \) a PPT algorithm that on input \( i \) outputs a uniformly random element of \( \mathcal{D}_i \)
- **Evaluation:** \( \exists \) PPT that on input \( i, x \in \mathcal{D}_i \) outputs \( f_i(x) \)
- **Hard to invert:** \( \forall \) n.u. PPT adversary \( \mathcal{A} \), \( \exists \) a negligible function \( \mu(\cdot) \) s.t.:

\[
\Pr [i \leftarrow \text{Gen}(1^n), x \leftarrow \mathcal{D}_i, y \leftarrow f_i(x) : f_i(\mathcal{A}(1^n, i, y)) = y] \leq \mu(n)
\]
Trapdoor Permutations

Definition (Trapdoor OWPs)

A collection of trapdoor permutations is a family of permutations \( \mathcal{F} = \{f_i : \mathcal{D}_i \rightarrow \mathcal{R}_i\}_{i \in \mathcal{I}} \) satisfying the following properties:

- **Sampling function:** \( \exists \) a PPT \( \text{Gen} \) s.t. \( \text{Gen}(1^n) \) outputs \( (i, t) \in \mathcal{I} \)
- **Sampling from domain:** \( \exists \) a PPT algorithm that on input \( i \) outputs a uniformly random element of \( \mathcal{D}_i \)
- **Evaluation:** \( \exists \) PPT that on input \( i, x \in \mathcal{D}_i \) outputs \( f_i(x) \)
- **Hard to invert:** \( \forall \) n.u. PPT adversary \( \mathcal{A} \), \( \exists \) a negligible function \( \mu(\cdot) \) s.t.:
  \[
  \Pr[i \leftarrow \text{Gen}(1^n), x \leftarrow \mathcal{D}_i, y \leftarrow f_i(x) : f_i(\mathcal{A}(1^n, i, y)) = y] \leq \mu(n)
  \]
- **Inversion with trapdoor:** \( \exists \) a PPT algorithm that given \( (i, t, y) \) outputs \( f_i^{-1}(y) \)
Public-key Encryption from Trapdoor Permutations

Let \( \mathcal{F} = \{ f_i : D_i \rightarrow R_i \}_{i \in \mathcal{I}} \) be a family of trapdoor permutations

Theorem (PKE from Trapdoor Permutations)

\((\text{Gen}, \text{Enc}, \text{Dec})\) is IND-CPA secure public-key encryption scheme
Public-key Encryption from Trapdoor Permutations

Let $\mathcal{F} = \{f_i : D_i \rightarrow R_i\}_{i \in \mathcal{I}}$ be a family of trapdoor permutations

- $\text{Gen}(1^n)$: $(f_i, f_i^{-1}) \leftarrow \text{Gen}_T(1^n)$. Output $(pk, sk) \leftarrow ((f_i, h_i), f_i^{-1})$

Theorem (PKE from Trapdoor Permutations)

$(\text{Gen, Enc, Dec})$ is IND-CPA secure public-key encryption scheme
Public-key Encryption from Trapdoor Permutations

Let $\mathcal{F} = \{f_i : D_i \rightarrow R_i\}_{i \in \mathcal{I}}$ be a family of trapdoor permutations

- $\text{Gen}(1^n)$: $(f_i, f_i^{-1}) \leftarrow \text{Gen}_T(1^n)$. Output $(pk, sk) \leftarrow ((f_i, h_i), f_i^{-1})$
- $\text{Enc}(pk, m)$: Pick $r \leftarrow \$ \{0, 1\}^n$. Output $(f_i(r), h_i(r) \oplus m)$

**Theorem (PKE from Trapdoor Permutations)**

$(\text{Gen}, \text{Enc}, \text{Dec})$ is IND-CPA secure public-key encryption scheme
Let $\mathcal{F} = \{f_i : \mathcal{D}_i \rightarrow \mathcal{R}_i\}_{i \in I}$ be a family of trapdoor permutations

- $\text{Gen}(1^n): (f_i, f_i^{-1}) \leftarrow \text{Gen}_T(1^n)$. Output $(pk, sk) \leftarrow ((f_i, h_i), f_i^{-1})$
- $\text{Enc}(pk, m)$: Pick $r \leftarrow \{0, 1\}^n$. Output $(f_i(r), h_i(r) \oplus m)$
- $\text{Dec}(sk, (c_1, c_2))$: $r \leftarrow f_i^{-1}(c_1)$. Output $c_2 \oplus h_i(r)$

**Theorem (PKE from Trapdoor Permutations)**

$(\text{Gen}, \text{Enc}, \text{Dec})$ is IND-CPA secure public-key encryption scheme
Let $\mathcal{F} = \{ f_i : D_i \rightarrow R_i \}_{i \in \mathcal{I}}$ be a family of trapdoor permutations

- **Gen($1^n$):** $(f_i, f_i^{-1}) \leftarrow \text{Gen}_T(1^n)$. Output $(pk, sk) \leftarrow ((f_i, h_i), f_i^{-1})$
- **Enc($pk, m$):** Pick $r \leftarrow \{0, 1\}^n$. Output $(f_i(r), h_i(r) \oplus m)$
- **Dec($sk, (c_1, c_2)$):** $r \leftarrow f_i^{-1}(c_1)$. Output $c_2 \oplus h_i(r)$

**Theorem (PKE from Trapdoor Permutations)**

$(\text{Gen}, \text{Enc}, \text{Dec})$ is IND-CPA secure public-key encryption scheme

**Think:** Proof?
Public-key Encryption from Trapdoor Permutations

Let $\mathcal{F} = \{f_i : D_i \rightarrow R_i\}_{i \in \mathcal{I}}$ be a family of trapdoor permutations

- **Gen**: $(f_i, f_i^{-1}) \leftarrow \text{Gen}_T(1^n)$. Output $(pk, sk) \leftarrow ((f_i, h_i), f_i^{-1})$

- **Enc**: $(pk, m)$: Pick $r \leftarrow \{0, 1\}^n$. Output $(f_i(r), h_i(r) \oplus m)$

- **Dec**: $(sk, (c_1, c_2))$: $r \leftarrow f_i^{-1}(c_1)$. Output $c_2 \oplus h_i(r)$

**Theorem (PKE from Trapdoor Permutations)**

$(\text{Gen}, \text{Enc}, \text{Dec})$ is IND-CPA secure public-key encryption scheme

- **Think**: Proof?

- **How to build trapdoor permutations?**
Candidate Trapdoor Permutations

**Definition (RSA Collection)**

\[ \text{RSA} = \{ f_i : \mathcal{D}_i \rightarrow \mathcal{R}_i \}_{i \in \mathcal{I}} \text{ where:} \]

- \[ \mathcal{I} = \{ (N, e) \mid N = p \cdot q \text{ s.t. } p, q \in \Pi_n, \ e \in \mathbb{Z}_{\Phi(N)}^* \} \]
- \[ \mathcal{D}_i = \{ x \mid x \in \mathbb{Z}_N^* \} \]
- \[ \mathcal{R}_i = \mathbb{Z}_N^* \]
- \[ \text{Gen}(1^n) \rightarrow ((N, e), d) \text{ where } (N, e) \in \mathcal{I} \text{ and } e \cdot d = 1 \mod \Phi(N) \]
- \[ f_{N,e}(x) = x^e \mod N \]
- \[ f_{N,d}^{-1}(y) = y^d \mod N \]
Candidate Trapdoor Permutations

Definition (RSA Collection)

\[ \text{RSA} = \{f_i : D_i \to R_i\}_{i \in I} \text{ where:} \]

- \[ I = \{(N, e) \mid N = p \cdot q \text{ s.t. } p, q \in \Pi_n, \ e \in \mathbb{Z}_\Phi(N)\} \]
- \[ D_i = \{x \mid x \in \mathbb{Z}_N^*\} \]
- \[ R_i = \mathbb{Z}_N^* \]
- \[ \text{Gen}(1^n) \to ((N, e), d) \text{ where } (N, e) \in I \text{ and } e \cdot d = 1 \mod \Phi(N) \]
- \[ f_{N, e}(x) = x^e \mod N \]
- \[ f_{N, d}^{-1}(y) = y^d \mod N \]

Think: Why is \( f_{N, e} \) a permutation?
Assumption (RSA Assumption)

For any n.u. PPT adversary \( A \), there exists a negligible function \( \mu(\cdot) \) s.t.:

\[
\Pr \left[ p, q \xleftarrow{\$} \Pi_n, \ N = p \cdot q, \ e \xleftarrow{\$} \mathbb{Z}_\Phi(N), \ y \xleftarrow{\$} \mathbb{Z}_N^*; \ x \xleftarrow{} A(N, e, y) : x^e = y \mod N \right] \leq \mu(n)
\]

Theorem

Assuming the RSA assumption, the RSA collection is a family of trapdoor permutations
Candidate Trapdoor Permutations (contd.)

Assumption (RSA Assumption)

For any n.u. PPT adversary $A$, there exists a negligible function $\mu(\cdot)$ s.t.:

$$\Pr \left[ \begin{array}{l}
    p, q \leftarrow \Pi_n, \ N = p \cdot q, \ e \leftarrow \mathbb{Z}_\Phi(N)^*, \\
    y \leftarrow \mathbb{Z}_N^*, \ x \leftarrow A(N, e, y) \\
    : \ x^e = y \mod N
\end{array} \right] \leq \mu(n)$$

- **Think:** RSA assumption implies the factoring assumption

Theorem

**Assuming the RSA assumption, the RSA collection is a family of trapdoor permutations**
Food for Thought

- Direct (more efficient) constructions of PKE (e.g., El-Gamal)
- Stronger security notions:
  - Indistinguishability under chosen-ciphertext attacks (IND-CCA) [Naor-Segev],[Dolev-Dwork-Naor],[Sahai]
  - Circular security/key-dependent message security [Boneh-Halevi-Hamburg-Ostrovsky]
  - Leakage-resilient encryption [Dziembowski-Pietrzak], [Akavia-Goldwasser-Vaikuntanathan]
- Weaker security notions:
  - Deterministic encryption [Bellare-Boldyreva-O’Neill]