Cryptography examples

Secret Key Encryption (SKE)

- **Perfect Secrecy**
  - ***perfectly secure:***
    * One-time Pad
    * One-time Pad only for first l bits, discarding the rest (but allows some error for Dec as it just chooses the rest randomly)
    * shift cypher for one bit
    * Vigenère Cypher for fixed key length that equals message length
    * monoalphabetic substitution cypher for messages with length ≤ 26
  - ***not perfectly secure:***
    * One-time Pad where *Enc* appends 0 and 1 with different probabilities
    * XOR mod 5 with $\mathcal{M} = \{0, \ldots, 4\}$ and $\mathcal{K} = \{0, \ldots, 5\}$
    * One-time Pad but excluding 0 as key
    * Vigenère Cypher for msg length n where we first choose uniformly the key length $\leq n$ and then the actual key.
    * all schemes with $|\mathcal{K}| < |\mathcal{M}|$
- **one-time Computational Security**
  - ***secure:***
    * $\text{Enc}(s, m) := G(s) \oplus m$ for secure PRG G
    * $\text{Enc}(k, m) := m \oplus F_k(0^n)$ for PRF F
  - ***not secure:***
    * $\text{Enc}(s, m) := G(s) \oplus m$ if G is not a secure PRG
    * $\text{Enc}(s, m) := (r, G(r) \oplus m)$ for PRG G
    * mode of operation $c_i := F_i(r + i + m_i)$ for $c_0 := r \leftarrow \$\{0, 1\}^n$ and PRP F
- **Pseudorandom Generators (PRG)**
  - ***secure:***
    * $G'(s) := G(s_1 \cdots s_{\lceil \lambda/2 \rceil})$ for a PRG $G : \{0, 1\}^\lambda \to \{0, 1\}^{2\lambda}$
    * $G(s) := f(s) \parallel h(s)$ for f OWP and h harc-core for f
    * $F_k(0^n)$ for $k \in \{0, 1\}^\lambda$ and PRF F
    * $G(s) := F_s(1) \parallel F_s(2) \parallel \cdots \parallel F_s(l)$ for a length-preserving PRF F

- ***not secure:***
  - $G : \{0,1\}^\lambda \to \{0,1\}^{\lambda+1}, s \mapsto s \parallel \oplus_{i,=1}^{\lambda} s_i$
  - $G'(s) := G(0^{|s|} \parallel s)$ for a PRG $G : \{0,1\}^\lambda \to \{0,1\}^{2\lambda}$
  - $G'(s) := G(s) \parallel G(s+1)$ for a PRG $G : \{0,1\}^\lambda \to \{0,1\}^{2\lambda}$
  - $G(x) := f(x) \parallel h(x)$ for OWF f and h hard-core for f

- **One-way Functions (OWF)**
  - ***one-way:***
    - for OWF f the construction $g(x_1, x_2) := (f(x_1), x_2)$ for $|x_1| = |x_2|$
    - for OWF f the construction $g(x_1, x_2) := (f(x_1), 0^{|x_2|})$ where $|x_1| = |x_2|$
    - $(\mathrm{Gen}, \mathrm{Samp}, H)$ for CRH (Gen, H)
    - probably: prime factorization (equal length), discrete log
    - computing square roots (if factoring is hard)
    - $f(x) := F_x(0^{|x|})$ for (length-preserving) PRP F
    - for OWF h the construction

$$f(x) := \begin{cases} 0^{|x|} & \text{if } x_{n/2,\dots,x_n} = 0^{n/2}, \\ h(x_1, \dots, x_{n/2})0^{n/2} & \text{else} \end{cases}$$

    - $g(x) := f(x) \parallel f(f(x))$ for OWF f
    - $g(x \parallel j) := (f(x), j, x_j)$ for OWF f (but reveals a bit of x)
    - $g(x) := f(x \parallel 0)$ for OWF $f : \{0,1\}^\lambda \to \{0,1\}^\lambda$
    - $g \circ f$ for OWP g, f
  - ***not one-way:***
    - $f(x, y) := F_x(y)$ for (length-preserving) PRP F
    - $f(y) := F_{0^{|y|}}(y)$ for (length-preserving) PRP F
    - $g(x) := f(f(x))$ for length-preserving OWF f (but secure if f is OWP!)
    - $H_s(x) := x \oplus \mathrm{pad}(s)$ for public padding-function pad

- **Hard-core Predicates**
  - *Goldreich-Levin:* Let $f : \{0,1\}^n \to \{0,1\}^n$ be a OWF and

$$g : \{0,1\}^{2n} \to \{0,1\}^{2n}, (x, r) \mapsto g(x, r) := (f(x), r)$$

    then g is a OWF and $h(x, r) := \langle x, r \rangle$ is hard-core for g.

- **Pseudorandom Functions (PRF)**
  - ***secure:***
    - $F'_k(x) := F_k(0 \parallel x) \parallel F_k(1 \parallel x)$ for a PRF $F : \{0,1\}^{2n} \to \{0,1\}^n$

* *Goldreich-Goldwasser-Micali (GGM) construction:* For a PRG $G : \{0,1\}^\lambda \to \{0,1\}^{2\lambda}, s \mapsto G(s) := (G_0(s), G_1(s))$, the family

$$F_k : \{0,1\}^\lambda \to \{0,1\}^\lambda, x = x_1 \cdots x_n \mapsto F_k(x) := G_{x_n}(G_{x_{n-1}}(\dots(G_{x_1}(k)\cdots))$$

is a PRF
* *3-round Feistel:* constructs PRP from PRF $F : \{0,1\}^\lambda \times \{0,1\}^n \to \{0,1\}^n$. Construction of a single round:

$$\Psi_F : \{0,1\}^\lambda \times \{0,1\}^{2n} \to \{0,1\}^{2n}, (k, x, y) \mapsto (y, x \oplus F_k(y))$$

then using $x' := y, y' := x \oplus F_k(y)$
* $\mathcal{F}(\mathcal{H})$ for PRF $\mathcal{F}$ and almost universal $\mathcal{H}$
* $F_k(x) := H(k \parallel x)$ for Random-Oracle H
* $F'_{k_1,\dots,k_n}(x) := F_{k_1}(x), \dots, F_{k_n}(x)$ for PRF F

– ***not secure:***
  * $F'_k(x) := F_k(0 \parallel x) \parallel F_k(x \parallel 1)$ for a PRF $F : \{0,1\}^{2n} \to \{0,1\}^n$
  * $F_{A,b} : \{0,1\}^n \to \{0,1\}^n, x \mapsto Ax + b$ for a $n \times n$ matrix A and n-bit vector b
  * $F_k(x) := k \oplus x$
  * AES (but used in practice instead of a PRP)
  * GGM construction for VIL

- **Chosen-Plaintext Security (CPA)**

  – ***secure:***
  * $Enc(k, m) := (r, F_k(r) \oplus m)$ for $r \leftarrow \$\{0,1\}^{n(\lambda)}, m \in \{0,1\}^{l(\lambda)}$ and PRP F
  * $Enc(k, m) := F_k(r \parallel m)$ for $r \leftarrow \$\{0,1\}^{\lambda/2}, m \in \{0,1\}^{\lambda/2}$ and PRP F
  * $Enc(k, m) := (r, m_1 \oplus F_k(r), m_2 \oplus F_k(r+1))$ for $m = m_1 \parallel m_2$ with $|m_1| = |m_2|$ and $r \leftarrow \$\{0,1\}^n$
  * CBC (Cipher block chaining): choose $c_0 := r \leftarrow \$\{0,1\}^n$ and compute $c_i := F_k(c_{i-1} \oplus m_i)$
  * CTR (Counter mode): choose $c_0 := r \leftarrow \$\{0,1\}^n$, and compute $c_i := F_k(r + i - 1) \oplus m_i$

  – ***not secure:***
  * all deterministic schemes
  * ECB (Electronic Code Box): every block m_i is encrypted separately by $F_k(m_i)$ for PRP F
  * modified CBC where $r_i = IV + i$ for $IV \leftarrow \$\{0,1\}^n$ instead of choosing r randomly each time.
  * CPA for PRP F where $l_{in}(\lambda)$ is too small, so overlap probability is too big (must be superlogarithmic to be secure, i.e. $l_{in}(\lambda) \in \omega(\log \lambda)$)

- **Message Authentication Code (MAC)**
  - ***secure:***
    - FIL: every PRF $\mathcal{F} = \{F_k : \{0,1\}^n \to \{0,1\}^l\}$
    - FIL: CBC-Mac - for PRF F the construction

      $$h_s(m_1, \ldots, m_t) := F_s(m_t \oplus F_s(m_{t-1} \oplus \cdots \oplus F_s(m_1) \cdots)),$$

      i.e. $\mathrm{Tag}_s(m_1, \ldots, m_t) = t_l$ for $t_0 := r = 0$ and $t_i := F_k(t_{i-1} \oplus m_i)$
    - FIL: XOR-MAC - for PRF $\mathcal{F}$ and almost XOR-universal $\mathcal{H}$

      $$\mathrm{Tag}_k(m) := (r, F_k(r) \oplus h_s(m)),$$

      where $r \leftarrow \$\{0,1\}$
    - VIL: for a FIL MAC Π' for messages of length n
      $(r, \tau_1, \ldots, \tau_d) \leftarrow \mathrm{Tag}_k(m_1, \ldots, m_d)$, where $\tau_i \leftarrow \mathrm{Tag}'_k(r \parallel l \parallel i \parallel m_i)$, $|m| = l < 2^{n/4}$, and $r \leftarrow \$\{0,1\}^{n/4}$
      (If necessary final block is padded with 0s; this is even strongly secure)
    - for secure MAC Π: $\mathrm{Tag}'_k(m) := m[1] \parallel \tau$ for $\tau \leftarrow \mathrm{Tag}_k(m)$
    - (not strongly secure) Π' for a secure MAC $\Pi = (\mathrm{Tag}, \mathrm{Vrfy})$:
      $\mathrm{Tag}'_k(m) := \mathrm{Tag}_k(m) \parallel 0$, $\mathrm{Vrfy}'_k(t \parallel b) := \mathrm{Vrfy}_k(t)$
    - (not if given Vrfy-Oracle access) Π' for secure MAC Π:
      $\mathrm{Tag}'_k(m) := (0, \tau, 0, 0)$ for $\tau \leftarrow \mathrm{Tag}_k(m)$
      $\mathrm{Vrfy}'_k(c, t, i, b)$: If $c = 0$ output 1 iff $\mathrm{Vrfy}_k(m, \tau) = 1$, if $c = 1$ output 1 iff $\mathrm{Vrfy}_k(m, \tau) = 1$ and $k_i = b$
  - ***not secure:***
    - $\mathrm{Tag}_k(m) := F_k(0 \parallel m_0) \parallel F_k(1 \parallel m_1)$ for PRF $F : \{0,1\}^{2n} \to \{0,1\}^n$, where $m = m_0 \parallel m_1$ and $|m_0| = |m_1| = n - 1$
    - $\mathrm{Tag}_k(m_1, \ldots, m_l) := F_k(\langle 1 \rangle \parallel m_1) \oplus \cdots \oplus F_k(\langle l \rangle \parallel m_l)$ for PRF F and $m_i \in \{0,1\}^{n/2}$
    - $\mathrm{Tag}_k(m_1, \ldots, m_l) := (r, F_k(r) \oplus F_k(\langle 1 \rangle \parallel m_1) \oplus \cdots \oplus F_k(\langle l \rangle \parallel m_l))$ for PRF F, $m_i \in \{0,1\}^{n/2}$ and $r \leftarrow \$\{0,1\}^n$
    - $\mathrm{Tag}_k(m) := F_k(m_1) \parallel F_k(F_k(m_2))$ for PRF F and $m = m_0 \parallel m_1$ with $|m_0| = |m_1| = n$
    - CBC-MAC as VIL-MAC or as FIL-MAC with $r \neq 0^n$ or if all blocks are output.
    - VIL: CBC-MAC where message length is appended at the end of the message
    - VIL: $\mathrm{Tag}_{s,k}(m) := H^s(k \parallel m)$ for CRH H when H is constructed via Merkle-Damgård
    - CBC-Mac with $F_k(m) := F'_k(m) \parallel \langle i \rangle$ where $F'_k : \{0,1\}^n \to \{0,1\}^{n/2}$ is a secure MAC and $\langle i \rangle$ the $n/2$-bit encoding of the number of leading zeros of m

- **Chosen-Ciphertext Security (CCA)**
  - ***secure:***
    - all schemes that satisfy CPA+Auth
    - all Encrypt-then-Authenticate schemes with CPA secure Π and strongly secure MAC: (for any Π and only secure MAC still unforgeable) choose $(k_E, k_M) \leftarrow \$\{0,1\}^\lambda$, calculate $c \leftarrow \mathrm{Enc}_{k_E}(m)$, $\tau \leftarrow \mathrm{Tag}_{k_M}(c)$ and output (c, τ)
  - ***not secure:***
    - CBC and CTR modes of operation
- **Hash Functions**
  - ***collision resistant:***
    - $H^{s_1,s_2}(x) := H_1^{s_1}(x) \parallel H_2^{s_2}(x)$ where at least one of the hash functions H_1, H_2 is collision resistant
    - $H \circ H$ for CRH H
    - $H_s(x) := x \oplus \mathrm{pad}(s)$ for public padding-function pad
    - bootstrap construction from CRH with small compression by *Merkle-Damgård* or *Merkle Tree*
  - ***not collision resistant:***
    - VIL Merkle-Damgård (but can be strengthened to be CRH by including the length of the input)
    - for a CRH $\hat{h} : \{0,1\}^{2n-1} \to \{0,1\}^{n-1}$ the construction $h : \{0,1\}^{2n} \to \{0,1\}^n$ with

      $$h^s(0 \parallel x) = 0 \parallel \hat{h}^s(x) \text{ and } h^s(1 \parallel x) = 1^n$$

      (but the Merkle-Damgård transform using h is CRH!)

Public Key Encryption (PKE)

Note: for PKE it is allowed that Dec fails with negligible probability

- **Diffie-Hellman Key Excange (DDH/CDH)**
  1. generate parameters $(\mathbb{G}, g, p) \leftarrow \$\mathrm{GroupGen}(1^\lambda)$
  2. A chooses $x \leftarrow \$\mathbb{Z}_p^*$ and sends $g^x \mod p$ to B,
     B chooses $y \leftarrow \$\mathbb{Z}_p^*$ and sends $g^y \mod p$ to A
  3. A and B calculate the shared key $g^{xy} \mod p$
- **PKE constructions**
  - **secure PRG:**
    - (assuming DDH holds for all $t(\lambda) = \mathrm{poly}(\lambda)$) $(\mathbb{G}, g, p) \leftarrow \$\mathrm{GroupGen}(1^\lambda)$

      $$G_{g,q} : \mathbb{Z}_q^{t+1} \to \mathbb{G}^{2t+1}, (x, y_1, \dots, y_t) \mapsto (g^x, g^{y_1}, g^{xy_1}, \dots, g^{y_t}, g^{xy_t})$$

- **secure PRF:**
  - (under DDH assumption) *Naor-Reingold:* $(\mathbb{G}, g, p) \leftarrow \$\text{GroupGen}(1^\lambda)$

$$F_{q,g,\vec{a}} : \{0,1\}^n \to \mathbb{G}, (x_1, \ldots, x_n) \mapsto (g^{a_0})^{\Pi_{i=1}^n a_i^{x_i}}$$

- **CRH:**
  - (under DL assumption) $(\mathbb{QR}_p, g_1, p = 2q+1) \leftarrow \$\text{GroupGen}(1^\lambda)$, for $g_2 \leftarrow \$\mathbb{QR}_p$

$$H_{g_1,g_2,p,q} : \mathbb{Z}_q^2 \to \mathbb{QR}_p, (x_1, x_2) \mapsto g_1^{x_1} g_2^{x_2}$$

  - (assume RSA is hard relative to GenRSA) (Gen, H) for
    $\text{Gen}(1^\lambda) = (N, e, y) =: s$ for $(N, e) \leftarrow \text{GenRSA}(1^\lambda)$, $y \leftarrow \$\mathbb{Z}_N^*$

$$H^s : \{0,1\}^{3n} \to \mathbb{Z}_N^*, x \mapsto f_{x_{3n}}^s \left(f_{x_{3n-1}}^s (\cdots (1) \cdots) \right)$$

    for $f_0^s(x) := x^e \mod N$, $f_1^s(x) := yx^e \mod N$
  - (under DL assumption) $(\mathbb{G}, q, h_1) \leftarrow \$\text{GroupGen}(1^\lambda)$, $h_2, \ldots, h_t \leftarrow \mathbb{G}$
    $\text{Gen}(1^\lambda) \to (\mathbb{G}, q, (h_1, \ldots, h_t))$
    $H^s : \mathbb{Z}_q^t \to \mathbb{G}, (x_1, \ldots, x_t) \mapsto \Pi_i h_i^{x_i}$
- **hard-core predicate:**
  - least-significant bit for RSA and Rabin TDP
  - $\text{half}(x) := \begin{cases} 0 & \text{if } 0 < x < N/2 \\ 1 & \text{if } N/2 < x < N \end{cases}$ for the RSA problem

- **Trapdoor Permutation (TDP)**
  let $n = pq$ (p, q distinct, odd primes)

  - *Rivest, Shamir, Adleman (RSA):* (assuming factoring is hard) let e any value s.t. $\text{gdc}(e, \phi(n)) = 1$
    GenRSA $\to (pk, sk)$ with $pk = (n, e)$, $sk = (n, d)$ and $de = 1 \mod \phi(n)$
    $f_e(x) := x^e \mod n$, $f_d^{-1}(y) := y^d \mod n$
  - *Rabin:* (EQUIVALENT to hardness of factoring) $p, q \equiv 3 \mod 4$ (ensures, that f is a permutation on $\mathbb{QR}_n$)
    GenModulus $\to (pk, sk)$ with $pk = n$, $sk = (p, q)$
    $f : \mathbb{QR}_n \to \mathbb{QR}_n, x \mapsto x^2 \mod n$
    f^{-1} computes square-roots using Chinese Remainder Theorem and sk

- **CPA**

  - ***secure:***
    - $\Pi = (\text{KGen}, \text{Enc}, \text{Dec})$ for TPD (Gen, f, f^{-1}) and h hard-core for f:
      $\text{KGen}(1^\lambda) = \text{Gen}(1^\lambda)$
      $\text{Enc}(pk, m \in \{0,1\}) \to (f(pk, r), h(pk, r) \oplus m)$, $r \leftarrow \$\mathcal{X}_{pk}$
      $\text{Dec}(sk, (c_1, c_2)) := f^{-1}(sk, c_1) = r$, $m = h(pk, r) \oplus c_2$

* the above construction for RSA or Rabin TDP with $h :=$ lsb and the constraint $\text{lsb}(r) = m$ (assuming RSA/Rabin assumption holds)
* *modified RSA:* (see RSA TDP) let $l \in \omega(\log \lambda)$
GenRSA $\to (pk, sk)$ with $pk = (n, e)$, $sk = (n, d)$ ($de = 1 \mod \phi(n)$)
$\text{Enc}(pk, m) \to (m \parallel r)^e \mod n$, for $r \leftarrow \$\{0,1\}^l$
$\text{Dec}(sk, c) = c^d \mod n = m \parallel r$, output m
(standardized padding: $\hat{m} = 0 \parallel 1 \parallel r \parallel m$, $r \geq 8$ bytes)
* multiple encryption from a CPA secure scheme
($\text{Enc}'(pk, m) := \text{Enc}(pk, m_1) \dots \text{Enc}(pk, m_n)$)
* for single bit messages a variant of El Gamal:
$$\text{Enc}(pk, b) \to \begin{cases} (g^y, h^y) \text{ for } y \leftarrow \$\mathbb{Z}_q & \text{if } b = 0 \\ (g^y, g^z) \text{ for } y, z \leftarrow \$\mathbb{Z}_q & \text{if } b = 1 \end{cases}$$
$\text{Dec}(sk, (c_1, c_2))$ output 0 if $c_1^x = c_2$, else 1
* *El Gamal:* (assuming DDH is hard relative to GroupGen)
$\text{Gen}(1^n)$: obtain $(\mathbb{G}, g, q) \leftarrow$ GroupGen with g generator, $x \leftarrow \$\mathbb{Z}_q$ and $h := g^x$, set $pk := (\mathbb{G}, g, q, h)$, $sk = (\mathbb{G}, g, q, x)$
$\text{Enc}(pk, m) \to (g^y, h^y \cdot m)$ for $y \leftarrow \$\mathbb{Z}_q$
$\text{Dec}(sk, (c_1, c_2)) = c_2 / c_1^x$

– ***not secure:***
  * RSA for $l \in \mathcal{O}(\log \lambda)$ or without padding
  * any scheme that outputs cyphertexts c with $|c| \in \mathcal{O}(\log \lambda)$
  * a variant of El Gamal with $\mathbb{G} := \mathbb{QR}_p$, $p = 2q + 1$ and $\text{Enc}(pk, m) \to (g^r, h^r + m)$ for $r \leftarrow \$\mathbb{Z}_q$

• **CCA**

– ***secure:***
  * **(CCA-1)** *Cramer-Shoup Lite:* (assuming DDH is hard)
$\text{Gen}(1^n)$: obtain params $:= (\mathbb{G}, g_1, g_2, q) \leftarrow$ GroupGen with g_1, g_2 generators, $x_i, y_i \leftarrow \$\mathbb{Z}_q$ and $h_i := g_1^{x_i} g_2^{y_i}$ for $i = 1, 2$.
Set $pk := (\text{params}, h_1, h_2)$, $sk = (x_1, y_1, x_2, y_2)$
$\text{Enc}(pk, m) \to (g_1^r, g_2^r, h_1^r \cdot m, h_2^r)$ for $r \leftarrow \$\mathbb{Z}_q$
$$\text{Dec}(sk, (c_1, c_2, c_3, c_4)) := \begin{cases} c_3 / (c_1^{x_1} c_2^{y_1}) & \text{if } c_4 = c_1^{x_2} c_2^{y_2} \\ \perp & \text{else} \end{cases}$$
  * **(CCA-2)** *Cramer-Shoup:* (assuming DDH is hard)
$\text{Gen}(1^n)$: obtain params $:= (\mathbb{G}, g_1, g_2, q) \leftarrow$ GroupGen with g_1, g_2 generators, $x_i, y_i \leftarrow \$\mathbb{Z}_q$ and $h_i := g_1^{x_i} g_2^{y_i}$ for $i = 1, 2, 3$.
For CRH H set $pk := (\text{params}, h_1, h_2, h_3, H)$, $sk = (x_1, y_1, x_2, y_2, x_3, y_3)$
$\text{Enc}(pk, m) \to (g_1^r, g_2^r, h_1^r \cdot m, (h_2 h_3^\beta)^r)$ for $r \leftarrow \$\mathbb{Z}_q$ and $\beta := H(c_1, c_2, c_3)$
$$\text{Dec}(sk, (c_1, c_2, c_3, c_4)) := \begin{cases} c_3 / (c_1^{x_1} c_2^{y_1}) & \text{if } c_4 = c_1^{x_2 + \beta x_3} c_2^{y_2 + \beta y_3} \\ \perp & \text{else} \end{cases}$$

* **(CCA-2)** $\Pi = (\text{Setup}, \text{KGen}, \text{Enc}, \text{Dec})$ selective IND-ID-CPA IBE with ID space $\{0,1\}^n$ and $\Pi' = (\text{KGen}', \text{Sign}, \text{Vrfy})$ 1-time UF-CMA Signature
  $\text{KGen}''(1^\lambda) : (mpk, msk) \leftarrow \$\text{Setup}(1^\lambda)$, set $ek = mpk$, $dk = msk$
  $\text{Enc}''(ek, m)$: sample $(vk, sk) \leftarrow \$\text{KGen}'(1^\lambda)$ s.t. $|vk| = n(\lambda)$. Output $c'' := (c, vk, \sigma)$ for $c \leftarrow \$\text{Enc}(mpk, vk, m)$ $(ID = vk)$, $\sigma \leftarrow \$\text{Sign}(sk, c)$
  $\text{Dec}''(dk, c'' = (c, vk, \sigma))$: check $\text{Vrfy}(vk, c, \sigma) = 1$, if not return $\perp$ else return $\text{Dec}(d_{vk}, c)$ where $d_{vk} \leftarrow \$\text{KGen}(msk, vk)$

- ***not secure:***
  * RSA
  * multiple encryption from a CCA secure scheme
    $(\text{Enc}'(pk, m) := \text{Enc}(pk, m_1) \ldots \text{Enc}(pk, m_n))$
  * every malleable encryption scheme
  * El Gamal
  * CS-Lite is not CCA-2 secure

• **Digital Signatures**

- ***secure:***
  * *Full-Domain Hash:* (RO-model) for TPD (Gen, f, f^{-1})
    $\text{KGen}(1^\lambda) = \text{Gen}(1^\lambda)$
    $\text{Sign}(m, sk) = f^{-1}(sk, H(m))$ for RO H
    $\text{Vrfy}(pk, m, \sigma) = 1$ iff $f(pk, \sigma) = H(m)$
  * *Waters Signature:* (assuming CDH is hard)
    $\text{KGen}(1^\lambda)$: obtain params $:= (\mathbb{G}, \mathbb{G}_T, g, q, \hat{e}) \leftarrow \$\text{BilGroupGen}$ with g generator of $\mathbb{G}$, $a \leftarrow \$\mathbb{Z}_q$, $g_1 := g^a$, $g_2, u_0, \ldots, u_k \leftarrow \$\mathbb{G}$.
    Set $pk := (\text{params}, g_1, g_2, u_0, \ldots, u_k)$, $sk := g_2^a$
    $\text{Sign}(sk, m) \to \sigma := (g_2^a \cdot \alpha(m)^r, g^r)$ for $r \leftarrow \$\mathbb{Z}_q$, $\alpha(m) := u_0 \prod_{i=1}^{k} u_i^{m[i]}$ with $m = m[1] \ldots m[k]$
    $\text{Vrfy}(pk, m, (\sigma_1, \sigma_2))$: check $\hat{e}(g, \sigma_1) = \hat{e}(\sigma_2, \alpha(m)) \cdot \hat{e}(g_1, g_2)$
  * *Fiat-Shamir Transform:* (RO-model) for passively secure and canonical ID-scheme $\Pi = (\text{Gen}_{ID}, \mathcal{P}_1, \mathcal{P}_2, \mathcal{V})$
    $\text{Gen}(1^\lambda) = \text{Gen}_{ID}(1^\lambda)$ and RO $H : \{0,1\}^* \to \mathcal{B}_{pk,\lambda}$
    $\text{Sign}(sk, m) \to (\alpha, \gamma)$ for $(\alpha, s) \leftarrow \mathcal{P}_1(sk)$, $\beta := H(\alpha, m)$, $\gamma := \mathcal{P}_2(sk, s, \beta)$
    $\text{Vrfy}(pk, m, (\alpha, \gamma))$: output 1 iff $\tau := (\alpha, \beta, \gamma)$ for $\beta = H(\alpha, m)$ is a valid transcript
  * (RO model) all canonical ID schemes with special soundness and honest verifyer zero knowledge (HVZK) s.t. $|\mathcal{B}_{pk,\lambda}| \in \omega(\log \lambda)$
  * **(only 1-time)** for OWP f and $\mathcal{M} := \{1, \ldots, \lambda\}$:
    $\text{Gen}(1^\lambda)$: for $x, x' \leftarrow \$\{0,1\}^\lambda$ and $y := f^{(\lambda)}(x)$, $y' := f^{(\lambda)}(x')$
    set $pk := (x, x')$, $sk = (y, y')$
    $\text{Sign}(sk, i) := (f^{(\lambda - i)}(x), f^{(i)}(x'))$
    $\text{Vrfy}(pk, i, (\sigma, \sigma')) = 1$ iff $y = f^{(i)}(x)$ and $y' = f^{(\lambda - i)}(x')$
    where f^k denotes the k-times application of f and $f^{(0)} := \text{id}_{\{0,1\}^\lambda}$

* $\Pi = (\text{Setup}, \text{KGen}, \text{Enc}, \text{Dec})$ IND-ID-CPA IBE with $|\mathcal{M}_{IBE}| = \omega(\log \lambda)$
      $\text{KGen}'(1^\lambda) : (mpk, msk) \leftarrow \$\text{Setup}(1^\lambda)$, set $vk = mpk,\ sk = msk$
      $\text{Sign}(sk, m) \rightarrow \sigma := d_m$ for $d_m \leftarrow \$\text{KGen}(sk, ID = m)$
      $\text{Vrfy}(vk, m, \sigma)$: Let $m = ID$ and $\sigma = d_{ID}$, pick $\mu \leftarrow \$\mathcal{M}_{IBE}$ and encrypt $c \leftarrow \$\text{Enc}(ID, \mu)$. Check $\text{Dec}(\sigma = d_{ID}, c) = \mu$

  - ***not secure:***
    * RSA using $\text{Sign}(sk, m) := m^d \mod n$
    * (not even one-time) for OWP f and $\mathcal{M} := \{1, \dots, \lambda\}$:
      $\text{Gen}(1^\lambda) := (pk, sk) = (y, x)$ for $x \leftarrow \$\{0,1\}^\lambda$ and $y := f^{(\lambda)(x)}$
      $\text{Sign}(x, i) := f^{(\lambda - i)}(x)$
      $\text{Vrfy}(y, i, \sigma) = 1$ iff $y = f^{(i)}(x)$
      where f^k denotes the k-times application of f and $f^{(0)} := \text{id}_{\{0,1\}^\lambda}$

- **Identification Schemes**
  - ***passively secure:***
    * *Schnorr:* (assuming DL is hard) three-round ID scheme $\Pi = (\text{Gen}, \mathcal{P}_1, \mathcal{P}_2, \mathcal{V})$
      $\text{Gen}(1^\lambda)$: obtain params $:= (\mathbb{G}, g, q) \leftarrow \text{GroupGen}(1^\lambda)$ with g generator, $x \leftarrow \$\mathbb{Z}_q$ and $y = g^x$.
      Set $pk := (\text{params}, y)$, $sk = x$ and $\mathcal{B}_{pk,\lambda} = \mathbb{Z}_q$
      1. $\mathcal{P}_1(sk)$ chooses $a \leftarrow \$\mathbb{Z}_q$ and outputs $\alpha := g^a$ (state $s = (pk, sk, a)$)
      2. $\mathcal{V}$ sends $\beta \leftarrow \$\mathbb{Z}_q$ to $\mathcal{P}$
      3. $\mathcal{P}_2(sk, s, \beta)$ outputs $\gamma := \beta x + a$
      4. $\mathcal{V}$ checks if $g^\gamma \cdot y^{-\beta} = \alpha$

- **Identity-Based Encryption (IBE)**
  - ***selective IND-ID-CPA:***
    * (assuming DBDH is hard) $\Pi = (\text{Setup}, \text{KGen}, \text{Enc}, \text{Dec})$
      $\text{Setup}(1^\lambda)$: choose params $:= (\mathbb{G}, \mathbb{G}_T, g, q, \hat{e}) \leftarrow \$\text{BilGroupGen}(1^\lambda)$ with g generator of $\mathbb{G}$, $\alpha \leftarrow \$\mathbb{Z}_q$, $h, g_2 := g^\beta \leftarrow \$\mathbb{G}$ and $g_1 := g^\alpha$.
      Set $mpk := (\text{params}, g_1, g_2, h)$, $msk = g_2^\alpha$
      $\text{KGen}(msk, ID \in \mathbb{Z}_q)$: pick $r \leftarrow \$\mathbb{Z}_q$ and set $d_{ID} := (g_2^\alpha \cdot F(ID)^r, g^r)$ for $F : \mathbb{Z}_q \rightarrow \mathbb{G}, ID \mapsto g_1^{ID} \cdot h$
      $\text{Enc}(ID, m \in \mathbb{G})$: pick $\gamma \leftarrow \$\mathbb{Z}_q$ and output

      $$c = (u, v, w) := (\hat{e}(g_1, g_2)^\gamma \cdot m, g^\gamma, F(ID)^\gamma)$$

      $$\text{Dec}(d_{ID} = (d_0, d_1), c = (u, v, w)) = \frac{u \cdot \hat{e}(d_1, w)}{\hat{e}(v, d_0)}$$