

VERNAM CIPHER

$$C = \text{Enc}_k(P) = P \oplus k \quad P = \text{Dec}_k(C) = C \oplus k$$

CESAR CIPHER

$$C = \text{Enc}_k(P) = P \boxplus k \quad P = \text{Dec}_k(C) = C \boxminus k$$

CIRULAR SHIFT CIPHER

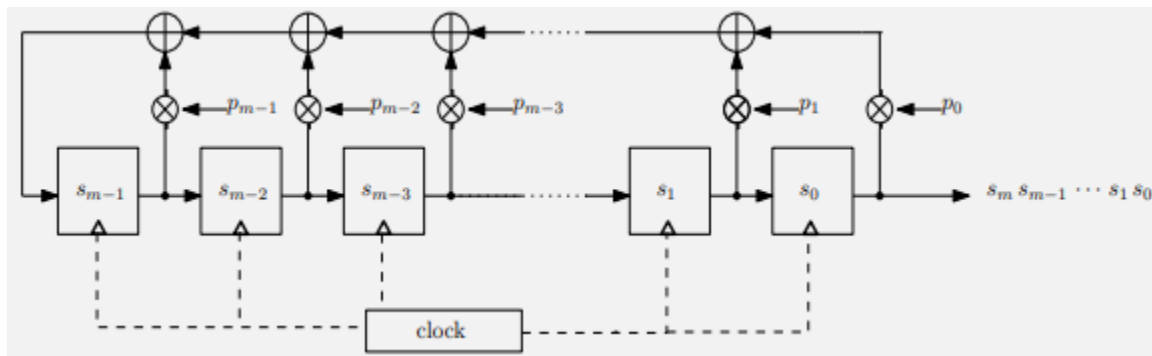
$$C = \text{Enc}_k(P) = P \lll k \quad P = \text{Dec}_k(C) = C \lll -k$$

LEHMER GENERATOR AND LCG

$$s_0 = \text{seed}$$

$$s_{i+1} \equiv a \cdot s_i + b \pmod{m}, \quad i = 0, 1, \dots$$

FIBONACCI LFSR



$$P(x) = 1 + p_{m-1}x + \dots + p_1x^{m-1} + p_0x^m$$

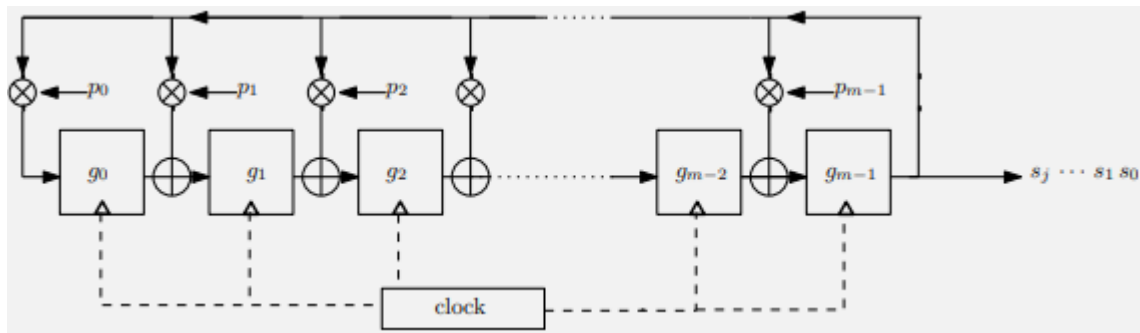
$$\chi_L(x) = x^m P\left(\frac{1}{x}\right) = p_0 + p_1x + \dots + p_{m-1}x^{m-1} + x^m$$

$$L = \begin{bmatrix} p_{m-1} & 1 & 0 & 0 & \dots & 0 \\ p_{m-2} & 0 & 1 & 0 & \dots & 0 \\ p_{m-3} & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_1 & 0 & 0 & 0 & \dots & 1 \\ p_0 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

$$\mathbf{s} = [s_{m-1} s_{m-2} s_{m-3} \dots s_1 s_0]$$

$$\mathbf{s} \cdot L = \mathbf{s}'$$

GALOIS LFSR

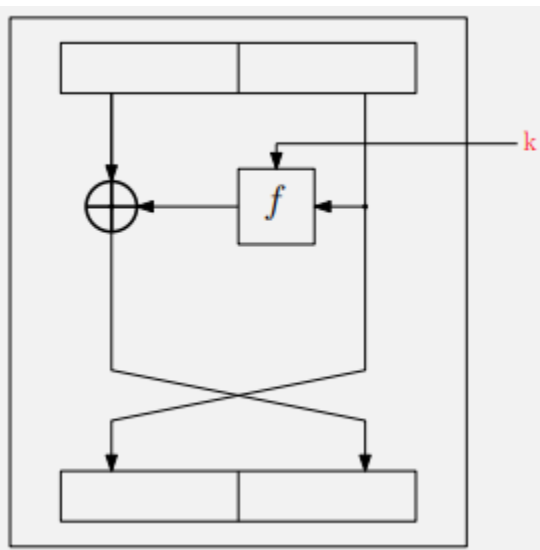


$$L = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \\ p_0 & p_1 & p_2 & p_3 & \cdots & p_{m-1} \end{bmatrix}$$

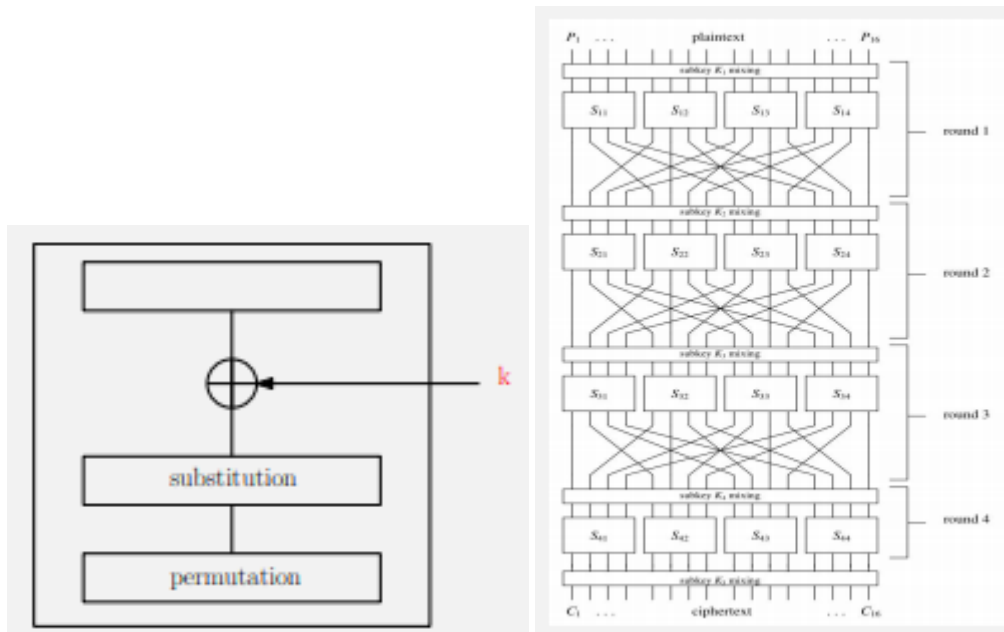
$$\mathbf{s} = [g_0 g_1 \cdots g_{m-2} g_{m-1}]$$

$$\mathbf{s} \cdot L = \mathbf{s}'$$

FEISTEL ROUND



SUBSTITUTION-PERMUTATION NETWORK



CBC CIPHERING

$y_1 = \text{Enc}_k(B1 \oplus IV)$
 If $j > 1$ then $y_j = \text{Enc}_k(Bj \oplus y_{j-1})$.

CBC DECIPHERING

$B1 = \text{Dec}_k(y_1) \oplus IV$
 If $j > 1$ then $Bj = \text{Dec}_k(y_j) \oplus y_{j-1}$.

CTR CIPHERING

$O_j = \text{Enc}_k(Tj)$ for $j = 1, \dots, N$
 $C_j = Bj \oplus O_j$ for $j = 1, \dots, N$

CTR DECIPHERING

$O_j = \text{Enc}_k(Tj)$ for $j = 1, \dots, N$
 $Bj = C_j \oplus O_j$ for $j = 1, \dots, N$

GCM TAG T

$H = \text{Enc}_k(0);$
 $g_0 = \text{AAD} \otimes H$; where \otimes is the Galois multiplication in $\text{GF}(2^{128})$.
 $g_j = (g_{j-1} \oplus C_j) \otimes H$ for $j = 1, \dots, N$;
 $t = (g_N \otimes H) \oplus \text{Enc}_k(T0)$.

EULER'S CRITERION

$$r^{\frac{p-1}{2}} \equiv 1 \pmod{p}$$

EULER'S FUNCTION

$$N = p^r \times q^s \times \dots, \phi(N), \mathbb{Z}_N^*, \phi(N) = \phi(p^r) \times \phi(q^s) \times \dots$$

$$\phi(p^r) = (p-1) \times p^{r-1}$$

FERMAT'S LITTLE THEOREM

$$r^p \equiv r \pmod{p}$$

CRT

$$\begin{cases} x = a_1 \bmod n_1 \\ x = a_2 \bmod n_2 \\ x = a_3 \bmod n_3 \end{cases} \quad N = n_1 * n_2 * n_3, \quad N_1 = n_2 * n_3, \quad N_2 = n_1 * n_3, \quad N_3 = n_1 * n_2$$

$$\begin{cases} N_1 * y_1 = 1 \bmod n_1 \\ N_2 * y_2 = 1 \bmod n_2 \\ N_3 * y_3 = 1 \bmod n_3 \end{cases} \quad x = a_1 * N_1 * y_1 + a_2 * N_2 * y_2 + a_3 * N_3 * y_3 \pmod{N}$$

$$f(a,b) = a*f(1,0) + b*f(0,1)$$

HMAC

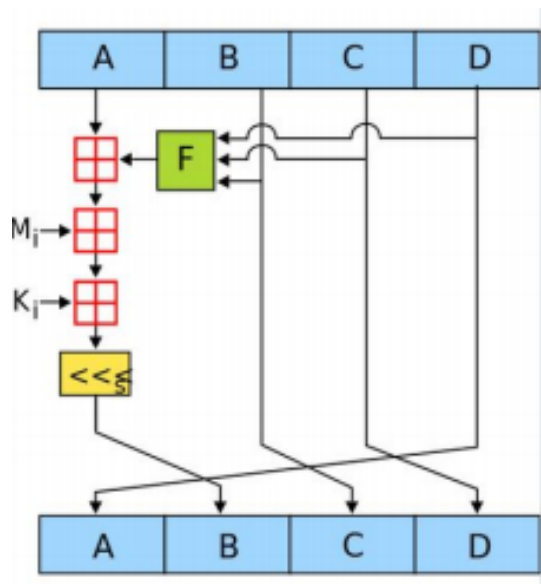
$$k^+ = \underbrace{00 \dots 00k}_{\text{blocksize}} \text{ and } \begin{cases} \text{ipad} = 0x36 \times \text{blocksize} \\ \text{opad} = 0x5c \times \text{blocksize} \end{cases}$$

$$\text{HMAC}_k(x) = h[k^+ \oplus \text{ipad} || h[(k^+ \oplus \text{opad}) || x]]$$

DAVID-MEYER FUNCTION COM

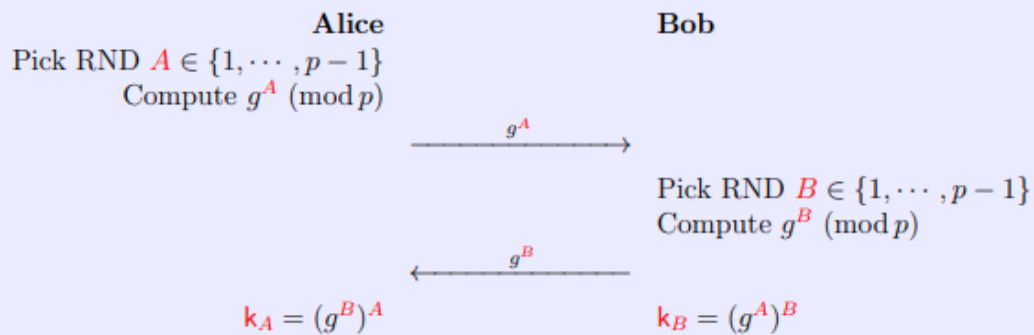
$$H_i = \text{Enc}_{x_i}(H_{i-1}) \oplus H_{i-1}$$

MD4 COMPRESSION



DIFFIE-HELMAN

Alice and **Bob** agree to use a prime number p and an element g of $\text{GF}^*(p)$.



So $k = k_A = k_B$ is the session key between **Alice** and **Bob**.

A and B are the private keys whilst g^A, g^B are the public keys.

ELGAMAL

- Parameter domains: g and p as in DH key agreement.
- $\text{Gen}(\lambda)$: pick $A \in \{1, \dots, p-1\}$ and compute $h = g^A$. Set $\text{pk} = h$ and $\text{sk} = A$.
- $\text{Enc}_{\text{pk}}(m)$ with $m \in \text{GF}(p)$: pick RND $B \in \{1, \dots, p-1\}$ and compute $C = (g^B, m \cdot h^B)$.
- $\text{Dec}_{\text{sk}}(C)$ with $C = (c_1, c_2)$: compute $m = c_2 / c_1^A$.

RABIN TEXTBOOK

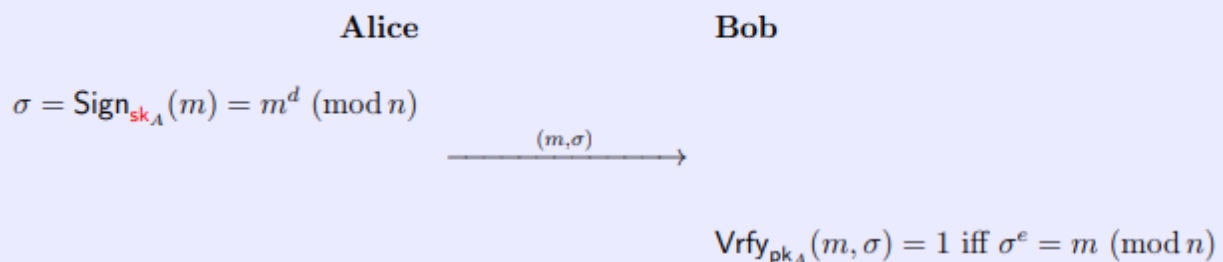
- $\text{Gen}(\lambda)$: choose $p, q \equiv 3 \pmod{4}$ (prime numbers of λ bit), compute $N = pq$.
- Set $\text{pk} = N$ and $\text{sk} = (p, q)$.
- $\text{Enc}_{\text{pk}}(m)$ with $m \in \mathbb{Z}_N$: compute $C = m^2 \pmod{N}$.
- $\text{Dec}_{\text{sk}}(C)$: compute $m_p = C^{\frac{p+1}{4}} \pmod{p}$, $m_q = C^{\frac{q+1}{4}} \pmod{q}$.
CRT gives four candidates for m : $(\pm m_p, \pm m_q)$.

RSA TEXTBOOK

- $\text{Gen}(\lambda)$: choose p, q (prime numbers of λ bit), compute $N = pq$ and $\phi(N) = (p-1)(q-1)$. Pick $e \in \mathbb{Z}_{\phi(N)}^*$ and compute $d = e^{-1} \pmod{\phi(N)}$. Set $\text{pk} = (N, e)$ and $\text{sk} = (\phi(N), d)$.
- $\text{Enc}_{\text{pk}}(m)$ with $m \in \mathbb{Z}_N$: compute $C = m^e \pmod{N}$.
- $\text{Dec}_{\text{sk}}(C)$: compute $m = C^d \pmod{N}$.

NAÏVE RSA-SIGNATURE

Alice's public key $\text{pk}_A = (n, e)$ and secret key is $\text{sk}_A = d$.



DSA – GEN

1. Generate a prime p with $2^{1023} < p < 2^{1024}$.
2. Find a prime divisor q of $p-1$ with $2^{159} < q < 2^{160}$.
3. Find an element α with $\text{ord}(\alpha) = q$, i.e., α generates the subgroup with q elements.
4. Choose a random integer d with $0 < d < q$.
5. Compute $\beta \equiv \alpha^d \pmod{p}$.

The keys are now:

$$k_{\text{pub}} = (p, q, \alpha, \beta)$$

$$k_{\text{pr}} = (d)$$

DSA -SIGN

1. Choose an integer as random ephemeral key k_E with $0 < k_E < q$.
2. Compute $r \equiv (\alpha^{k_E} \bmod p) \bmod q$.
3. Compute $s \equiv (SHA(x) + d \cdot r) k_E^{-1} \bmod q$.

DSA – VERIFY

1. Compute auxiliary value $w \equiv s^{-1} \bmod q$.
2. Compute auxiliary value $u_1 \equiv w \cdot SHA(x) \bmod q$.
3. Compute auxiliary value $u_2 \equiv w \cdot r \bmod q$.
4. Compute $v \equiv (\alpha^{u_1} \cdot \beta^{u_2} \bmod p) \bmod q$.
5. The verification $ver_{k_{pub}}(x, (r, s))$ follows from:

$$v \begin{cases} \equiv r \bmod q \implies \text{valid signature} \\ \not\equiv r \bmod q \implies \text{invalid signature} \end{cases}$$

POINT ADDITION

$$y^2 = x^3 + ax + b,$$

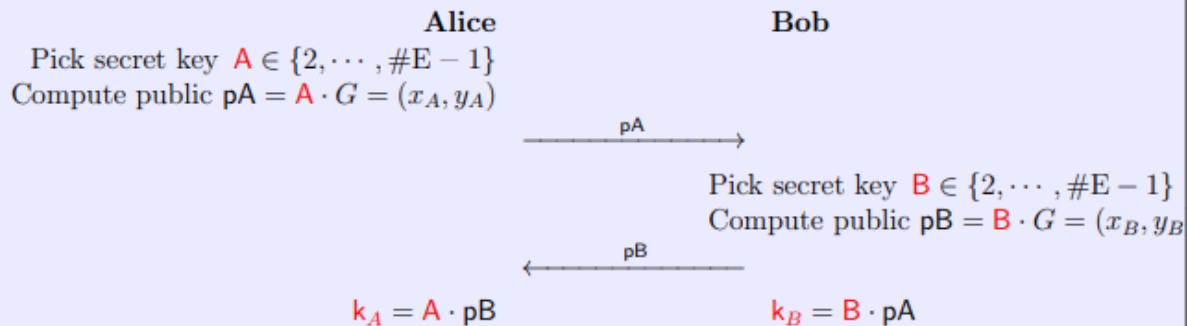
$$\begin{aligned} P + Q &= R \\ (x_p, y_p) + (x_q, y_q) &= (x_r, y_r) \end{aligned} \quad \begin{aligned} \lambda &= \frac{y_q - y_p}{x_q - x_p} \\ x_r &= \lambda^2 - x_p - x_q \\ y_r &= \lambda(x_p - x_r) - y_p \end{aligned}$$

POINT DOUBLING

$$\lambda = \frac{3x_p^2 + a}{2y_p}$$

EC-HD

Alice and **Bob** agree to use an elliptic curve E with parameters domain $T = (p, a, b, G, n, h)$.



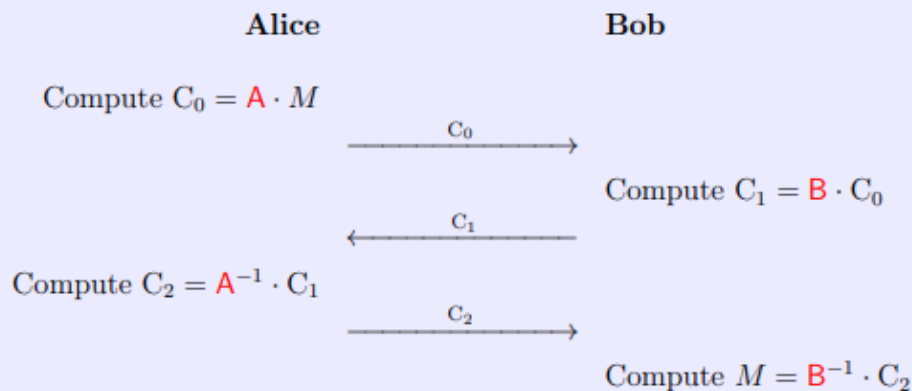
So $k = k_A = k_B$ is the session key between **Alice** and **Bob**.

A and B are the private keys whilst pA, pB are the public keys.

EC – MASSEY-OMURA

Alice has a secret key $0 < A < n$ such that $\gcd(A, n) = 1$. So she can efficiently compute $A^{-1} \pmod{n}$.

Bob has a secret key $0 < B < n$ such that $\gcd(B, n) = 1$. So he can efficiently compute $B^{-1} \pmod{n}$.



EC – DSA GEN

- 1) Choose a RND integer $d \in \{1, 2, \dots, n - 1\}$.
- 2) Compute $B = d \cdot G$.

The keys are:

$pk = (p, a, b, n, G, B);$

$sk = d$

EC – DSA SIGN

- 1) Choose an RND integer K as ephemeral key with $0 < K < n$.
- 2) Compute $R = K \cdot G = (x_R, y_R)$ and set $r = x_R$.
- 3) Compute $s = (\text{hash}(M) + d \cdot r) \cdot K^{-1} \pmod{n}$. If $s = 0 \pmod{n}$ go to 1).

The signature $\text{Sign}_{sk}(M)$ of M is the pair (r, s) .

EC – DSA VERIFY

- 1) Compute the auxiliar $w = s^{-1} \pmod{n}$.
- 2) Compute the auxiliar $u_1 = w \cdot \text{hash}(M) \pmod{n}$.
- 3) Compute the auxiliar $u_2 = w \cdot r \pmod{n}$.
- 4) Compute $P = u_1 \cdot G + u_2 \cdot B = (x_P, y_P)$.
- 5) The $\text{Vrfy}_{pk}(M, (r, s))$ outputs 1 for a valid signature if $x_P = r \pmod{n}$ otherwise $\text{Vrfy}_{pk}(M, (r, s))$ outputs 0.