VERNAM CIPHER

$$\mathrm{C} = \mathsf{Enc}_{\mathsf{k}}(\mathrm{P}) = \mathrm{P} \oplus \mathsf{k} \qquad \mathrm{P} = \mathsf{Dec}_{\mathsf{k}}(\mathrm{C}) = \mathrm{C} \oplus \mathsf{k}$$

CESAR CIPHER

$$\mathrm{C} = \mathsf{Enc}_{\mathsf{k}}(\mathrm{P}) = \mathrm{P} \boxplus \mathsf{k} \qquad \mathrm{P} = \mathsf{Dec}_{\mathsf{k}}(\mathrm{C}) = \mathrm{C} \boxplus \mathsf{k}$$

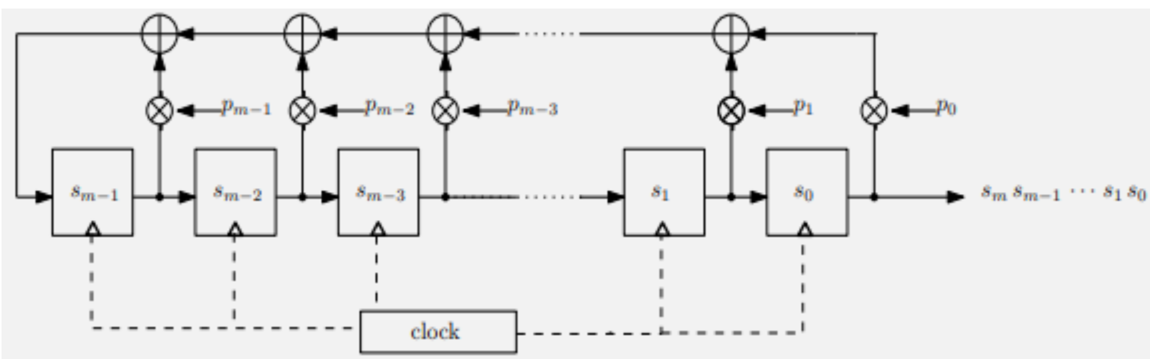
CIRULAR SHIFT CIPHER

$$\mathrm{C} = \mathsf{Enc}_{\mathsf{k}}(\mathrm{P}) = \mathrm{P} <<< \mathsf{k} \qquad \mathrm{P} = \mathsf{Dec}_{\mathsf{k}}(\mathrm{C}) = \mathrm{C} <<< -\mathsf{k}$$

LEHMER GENERATOR AND LCG

$$s_0 = \text{seed}$$
$$s_{i+1} \equiv a \cdot s_i + b \pmod{m}, \;\; i = 0, 1, \cdots$$

FIBONACCI LFSR



$$P(x) = 1 + p_{m-1}x + \cdots + p_1 x^{m-1} + p_0 x^m$$

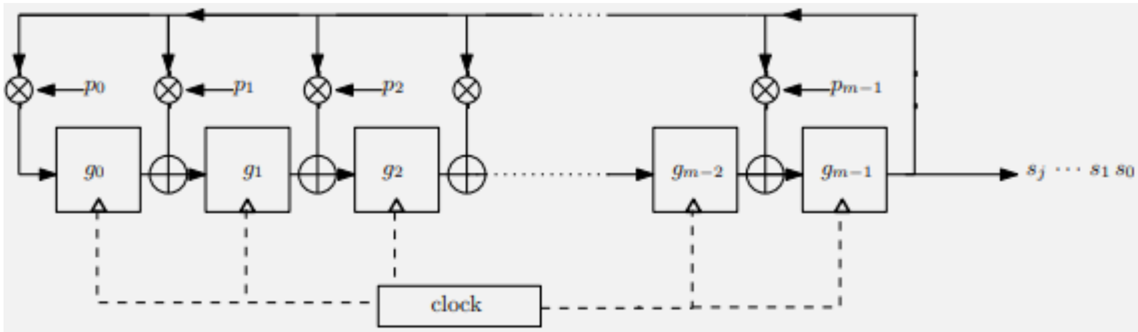
$$\chi_{\mathrm{L}}(x) = x^m P(\frac{1}{x}) = p_0 + p_1 x + \cdots + p_{m-1}x^{m-1} + x^m$$

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

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

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

GALOIS LFSR

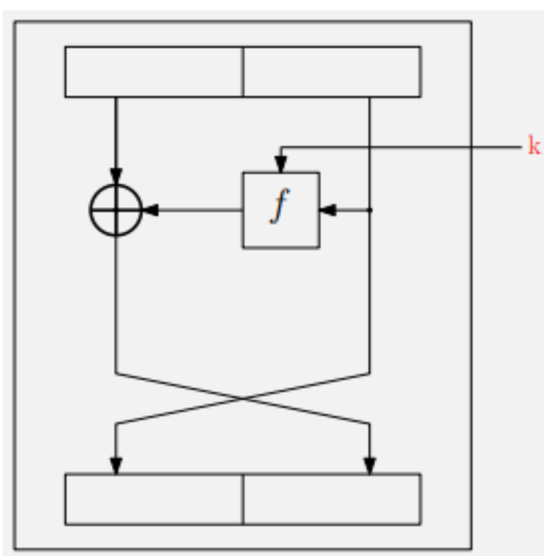


$$L = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \\ p_0 & p_1 & p_2 & p_3 & \cdots & p_{m-1} \end{bmatrix}$$

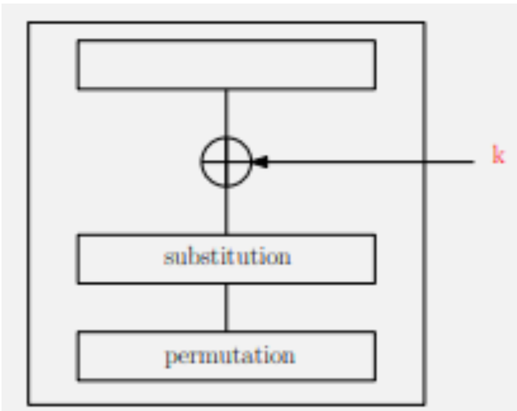
$$\mathbf{s} = [g_0 g_1 \cdots g_{m-2} g_{m-1}]$$

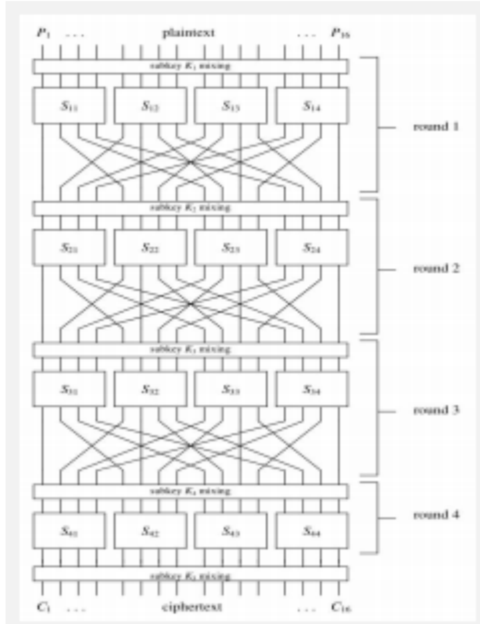
$$\mathbf{s} \cdot L = \mathbf{s}'$$

FEISTEL ROUND



SOBSTITUTION-PERMUTATION NETWORK





CBC CIPHERING

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

CBC DECIPHERING

$B1 = \mathsf{Dec}_{\mathsf{k}}(y_1) \oplus IV$
If $j > 1$ then $B_j = \mathsf{Dec}_{\mathsf{k}}(y_j) \oplus y_{j-1}$.

CTR CIPHERING

$\mathrm{O}_j = \mathsf{Enc}_{\mathsf{k}}(Tj)$ for $j = 1, \cdots, N$
$\mathrm{C}_j = Bj \oplus \mathrm{O}_j$ for $j = 1, \cdots, N$

CTR DECIPHERING

$\mathrm{O}_j = \mathsf{Enc}_{\mathsf{k}}(Tj)$ for $j = 1, \cdots, N$
$Bj = \mathrm{C}_j \oplus \mathrm{O}_j$ for $j = 1, \cdots, N$

GCM TAG T

$H = \mathsf{Enc}_{\mathsf{k}}(0)$;
$g_0 = \mathrm{AAD} \otimes H$; where $\otimes$ is the Galois multiplication in $\mathrm{GF}(2^{128})$.
$g_j = (g_{j-1} \oplus \mathrm{C}_j) \otimes H$ for $j = 1, \cdots, N$;
$\mathbf{t} = (g_N \otimes H) \oplus \mathsf{Enc}_{\mathsf{k}}(T0)$.

EULER'S CRITERION

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

EULER'S FUNCTION

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

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

FERMAT'S LITTLE THEOREM

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

CRT

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

$$\begin{cases} N_1 * y_1 = 1 \, mod \, n_1 \\ N_2 * y_2 = 1 \, mod \, n_2 \\ N_3 * y_3 = 1 \, mod \, n_3 \end{cases} \qquad x = a_1 * N_1 * y_1 + a_2 * N_2 * y_2 + a_3 * N_3 * y_3 \, (mod \, N)$$

---

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

HMAC

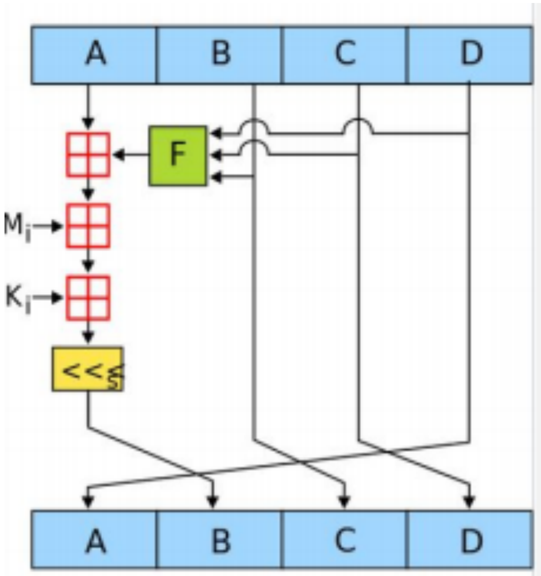
$$k^+ = \underbrace{00 \cdots 00k}_{blocksize} \text{ and } \begin{cases} \text{ipad} = \text{0x36} \times \text{blocksize} \\ \text{opad} = \text{0x5}c \times \text{blocksize} \end{cases}$$

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

DAVID-MEYER FUNCTION COM

$$H_i = Enc_{x_i}(H_{i-1}) \oplus H_{i-1}$$

MD4 COMPRESSION



DIFFIE-HELMAN

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

Alice		**Bob**
Pick RND $A \in \{1, \cdots, p-1\}$ Compute $g^A \pmod p$		
	$\xrightarrow{g^A}$	
		Pick RND $B \in \{1, \cdots, p-1\}$ Compute $g^B \pmod p$
	$\xleftarrow{g^B}$	
$\mathsf{k}_A = (g^B)^A$		$\mathsf{k}_B = (g^A)^B$

So $\mathsf{k} = \mathsf{k}_A = \mathsf{k}_B$ 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.
- $\mathsf{Gen}(\lambda)$: pick $A \in \{1, \cdots, p-1\}$ and compute $h = g^A$. Set $\mathsf{pk} = h$ and $\mathsf{sk} = A$.
- $\mathsf{Enc}_{\mathsf{pk}}(m)$ with $m \in GF(p)$: pick RND $B \in \{1, \cdots, p-1\}$ and compute $C = (g^B, m \cdot h^B)$.
- $\mathsf{Dec}_{\mathsf{sk}}(C)$ with $C = (c_1, c_2)$: compute $m = c_2 / c_1^A$.

RABIN TEXTBOOK

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

  CRT gives four candidates for m: $(\pm m_p, \pm m_q)$.

RSA TEXTBOOK

- $\mathsf{Gen}(\lambda)$: 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 $d = e^{-1} \bmod \phi(N)$. Set $\mathsf{pk} = (N, e)$ and $\mathsf{sk} = (\phi(N), d)$.
- $\mathsf{Enc}_{\mathsf{pk}}(m)$ with $m \in \mathbb{Z}_N$: compute $C = m^e \bmod N$.
- $\mathsf{Dec}_{\mathsf{sk}}(C)$: compute $m = C^d \bmod N$.

NAÏVE RSA-SIGNATURE

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

Alice **Bob**

$\sigma = \mathsf{Sign}_{\mathsf{sk}_A}(m) = m^d \pmod n$

$\xrightarrow{(m,\sigma)}$

$\mathsf{Vrfy}_{\mathsf{pk}_A}(m, \sigma) = 1$ iff $\sigma^e = m \pmod n$

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 $\mathrm{ord}(\alpha) = q$, i.e., α generates the subgroup with q elements.
4. Choose a random integer d with $0 < d < q$.
5. Compute $\beta \equiv \alpha^d \bmod p$.

The keys are now:

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

$k_{pr} = (d)$

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:

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

POINT ADDITION

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

$$\lambda = \frac{y_q - y_p}{x_q - x_p}$$

$$P + Q = R \qquad x_r = \lambda^2 - x_p - x_q$$

$$(x_p, y_p) + (x_q, y_q) = (x_r, y_r) \qquad y_r = \lambda(x_p - x_r) - y_p$$

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 = $(p\,,\,a\,,\,b\,,\,G\,,\,n\,,\,h)$.

Alice		**Bob**
Pick secret key $\mathsf{A} \in \{2, \cdots, \#\mathrm{E}-1\}$		
Compute public $\mathsf{pA} = \mathsf{A} \cdot G = (x_A, y_A)$		
	$\xrightarrow{\mathsf{pA}}$	
		Pick secret key $\mathsf{B} \in \{2, \cdots, \#\mathrm{E}-1\}$
		Compute public $\mathsf{pB} = \mathsf{B} \cdot G = (x_B, y_B$
	$\xleftarrow{\mathsf{pB}}$	
$\mathsf{k}_A = \mathsf{A} \cdot \mathsf{pB}$		$\mathsf{k}_B = \mathsf{B} \cdot \mathsf{pA}$

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

Alice		**Bob**
Compute $\mathrm{C}_0 = \mathsf{A} \cdot M$		
	$\xrightarrow{\mathrm{C}_0}$	
		Compute $\mathrm{C}_1 = \mathsf{B} \cdot \mathrm{C}_0$
	$\xleftarrow{\mathrm{C}_1}$	
Compute $\mathrm{C}_2 = \mathsf{A}^{-1} \cdot \mathrm{C}_1$		
	$\xrightarrow{\mathrm{C}_2}$	
		Compute $M = \mathsf{B}^{-1} \cdot \mathrm{C}_2$

EC – DSA GEN

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

The keys are:
$\mathsf{pk} = (p, a, b, n, G, B)$;
$\mathsf{sk} = d$

EC – DSA SIGN

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

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

EC – DSA VERIFY

1) Compute the auxiliar $w = s^{-1} \pmod{n}$.
2) Compute the auxiliar $u_1 = w \cdot \mathsf{hash}(M) \pmod{n}$.
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 $\mathsf{Vrfy}_{\mathsf{pk}}(M, (r, s))$ outputs 1 for a valid signature if $x_P = r \pmod{n}$ otherwise $\mathsf{Vrfy}_{\mathsf{pk}}(M, (r, s))$ outputs 0.