Crypt(?) -

Objectives
Requirements / Goals / Attributes / ...

CIA Triad	
	Confidentiality
	Integrity
	Availability

Authentication

Non-repudiation

Terminology

Cryptology *Cryptography* *Cryptanalysis* *Cryptogram*	*Attack* *Adversary* *Corrupted /* *Malicious Party*	*Defences* *Mitigations* *Countermeasures*

Pen & paper

Mechanical computation

Electronical computation

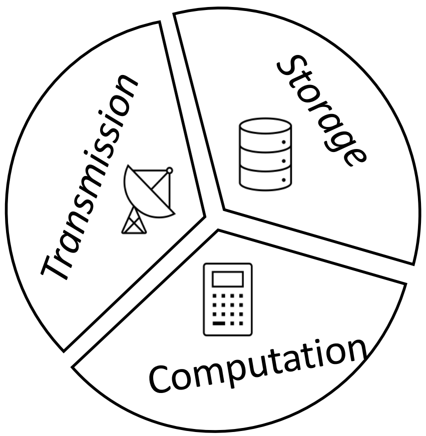
Quantum computation

Deteriorate

Corrupt

Impersonate

Attacks		
Passive	vs.	*Active*
Outsider		*Insider*



Secure all!

Alice Bob Charlie Daisy ... Oscar / Eve

Principles -

Kerckhoffs's principle

Only keep hidden the key.
(e.g., make the construction, and constants public)

Principle of (key) separation

Use different keys for different contexts, compartmentalize.
(e.g., minimise the damage of a leak)

Principle of diversity

Use different types of ... cryptographic algorithms.
(e.g., avoid same attacks against all)

Principle of simplicity

Keep everything simple.
(e.g., unnecessary complexity brings in risks)

Security by default

Keep default configuration as secure as possible.
(e.g., deny access by default)

Ethics!

Principle of minimal trust

Minimise the number of trusted entities, don't trust easily.
(e.g., do not say your secret to everyone)

Principle of the weakest link

A system cannot be more secure than its weaker component (link).
(e.g., secure all components)

Principle of least privilege

Grant the exact privileges required to perform the job.
(e.g., do not grand less or more privileges)

Security by design

Build in security from start.
(e.g., integrate security in the design and all the phases of the system)

Principle of modularization

Keep things modular.
(e.g., easily change one cipher with another)

Defence in depth

Use diverse security strategies at different layers.
(e.g., use physical and technical security)

Security by obscurity (?)

Oblivious Transfer, Obfuscation, Covert Channels, ... , Kleptography, Standardisation ...

Historical Ciphers -

Permutation | Substitution

... ciphers

Permutation

Rail Fence (variants)

```
v i t          v   a
a a             a i   n
r n              r     t
```

Key: 3
Plaintext: variant
Ciphertext: VITAARN / VAAINRT

Permutations (variants)

```
123 123 123 123
per mut ati onx
erp utm tia nxo
231 231 231 231
```

Key: (2,3,1)
Plaintext: permutation
Ciphertext: ERPUTMTIANXO

```
123     213
per     erp
mut     utm
ati     tia
onx     nxo
```

Key: (2,3,1)
Plaintext: permutation
Ciphertext: EUTNRTIXPMAO

Substitution

Monoalphabetic

Caesar Cipher (Shift Cipher)

```
a b c d e f g h i j k l m
C D E F G H I J K L M N O
n o p q r s t u v w x y z
P Q R S T U V W X Y Z A B
```

Plaintext: hello
Ciphertext: JGNNQ

No. of keys: 26 ⇨ Brute force

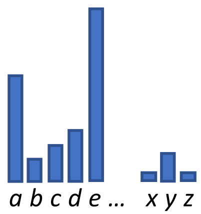
Simple Substitution

```
a b c d e f g h i j k l m
G M I P R S E N B X Z T A
n o p q r s t u v w x y z
L C V J F Y U D K H W O Q
```

Plaintext: hello
Ciphertext: NRTTC

No. of keys: 26!

Frequency analysis



Polyalphabetic

PlayFair (I=J)

```
C I P H E
R A B D F
G K L M N
O Q S T U
V W X Y Z
```

Key: CIPHER
Plaintext: This is a B
Ciphertext: YDPQPQBD

Vigenère cipher

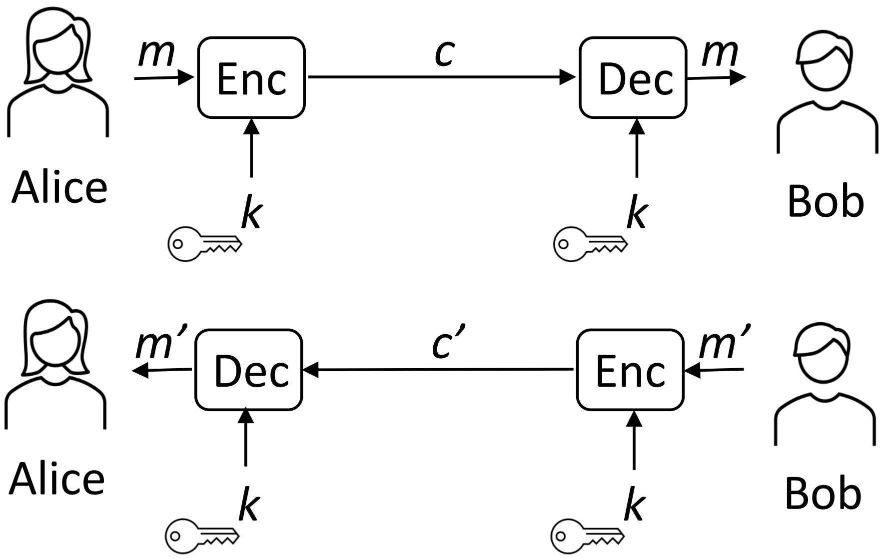
- Vigenère square
  26 Caesar alphabets

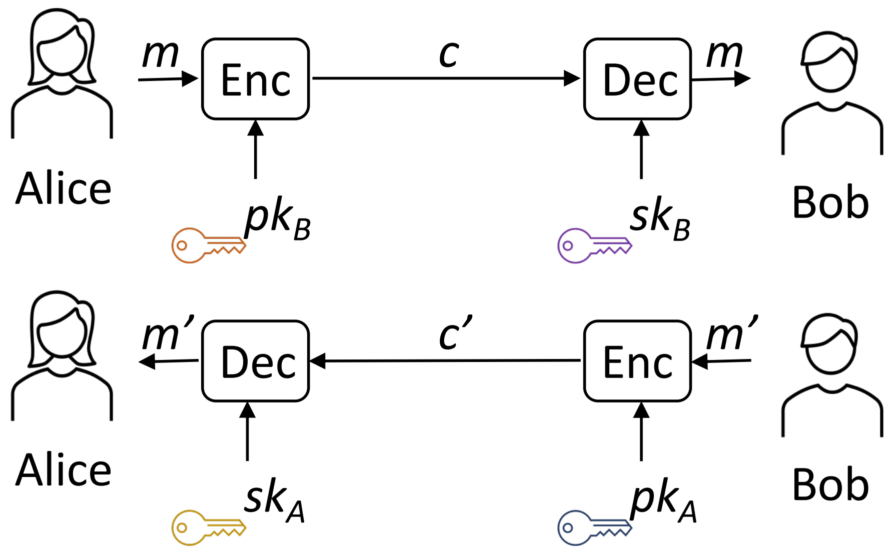
```
a b c d e f g h i ...
b c f e f g h i j ...
c d e f g h i j k ...
... ... ... ... ...
r s t u v w x y z ...
... ... ... ... ...
z a b c d e f g h ...
```

Key: CAR
Plaintext: Secret message
Ciphertext: UETTEK OEJUAXG

Symmetric vs. Asymmetric Encryption -

Symmetric	Asymmetric
... encryption	





Encryption: c = Enc(k,m)
Decryption: m=Dec(k,c)
Correctness:
Dec(k,Enc(k,m)) = m

Shorter keys +

Key distribution −

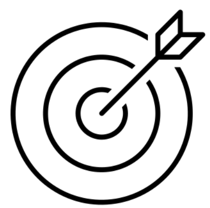
\+ Private keys never leave the owner

− Computational cost & speed

Encryption: c = Enc(pk_B,m)
Decryption: m=Dec(sk_B,c)
Correctness:
Dec(sk_B,Enc(pk_B,m)) = m

Terminology

k: symmetric key	m: plaintext
pk: public key	c: ciphertext
sk: private (secret) key	Enc: encryption alg.
(pk,sk): public-private key pair	Dec: decryption alg.
	Cryptanalysis



Confidentiality

No. of keys

for N bi-directional communicating parties

Each: N-1 [k]	vs.	Each: 1 [sk], N-1 [pk]
Total: N(N-1)/2 [k]		Total: N [sk], N [pk]

www.ruxandraolimid.weebly.com/pagesonsecurity

Pages on SecuRity by Ruxandra F. Olimid

Unconditional (Information-theoretical)	*Conditional (Computational)*

... security

*Provides security against an adversary with **no restrictions*** (e.g., unlimited computing power, time, memory)	*Provides security against an adversary with **computational restrictions*** (e.g., limited computing power, time, memory)
Stands against brute force (+)	(+) Suitable for practice
Good in theory, poor in practice (−)	(−) Weaker than unconditional security

For all m possible plaintext (i.e., in $\mathcal{M}$) and any c ciphertext (i.e., in $\mathcal{C}$) such that $Pr[C=c]>0$, it holds:

$$Pr[M=m|C=c] = Pr[M=m]$$

⇕

Perfect secrecy (Shannon 1949)

For all m_0, m_1 plaintexts of the same length (i.e., $|m_0| = |m_1|$) and for all c ciphertext, it holds:

$$Pr[Enc(k,m_0) = c] = Pr[Enc(k,m_1)=c]$$

where the key k is randomly chosen in the key space $\mathcal{K}$

Theorem (limitation):

Let (Enc, Dec) be a perfectly-secret encryption scheme over a plaintext space $\mathcal{M}$ and a key space $\mathcal{K}$. Then it holds that $|\mathcal{K}| \geqslant |\mathcal{M}|$ (i.e., the length of the key is larger or equal to the length of the message).

A scheme is **secure** if any adversary $\mathcal{A}$ that runs the attack in a time t succeeds the attack with probability at most ε.

Time t, probability ε can be:

- *Fixed*
- *Functions of a security parameter: n*

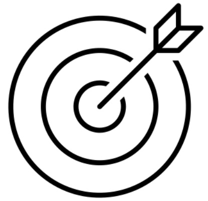
PPT(Probabilistic Polynomial in Time) Adversary:

- $t(n)$ is **polynomial** in n
- $\varepsilon(n)$ is **negligible** in n:

$\forall p(n), \exists n_d$ such that $\forall n \geqslant n_d$ it holds $\varepsilon(n) < 1/p(n)$

$p(n) = n^d$ and d constant

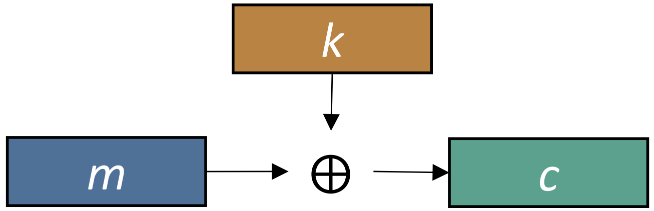
One Time Pad (OTP) -



Perfect secrecy

Vernam Cipher (1917)

Encryption: $c = k \oplus m$
Decryption: $m = k \oplus c$



The key *k:*

- is as long as the plaintext *m* and the ciphertext *c*
- is uniformly random chosen in $\mathcal{K}$

$$\begin{array}{rl} k\!: & 0\,1\,1\,0\,1\,1\,0\,0 \oplus \\ m\!: & 1\,0\,1\,1\,1\,0\,0\,1 \\ \hline c\!: & 1\,1\,0\,1\,0\,1\,0\,1 \end{array}$$

$$\begin{array}{rl} k\!: & \mathtt{B\ V\ Q\ G\ F\ B} \oplus \\ m\!: & \mathtt{N\ O\ T\ I\ M\ E} \pmod{26} \\ \hline c\!: & \mathtt{O\ J\ J\ O\ R\ F} \end{array}$$

Multiple use of the same key *k*

$c_1 = k \oplus m_1,\ c_2 = k \oplus m_2,\ c_3 = k \oplus m_3, \ldots$

1. *Ciphertext-only attack:* $\mathcal{A}$ just observes the ciphertexts

   $\mathcal{A}$ finds relations between plaintexts: $c_1 \oplus c_2 = m_1 \oplus m_2$

2. *Known-plaintext attack:* $\mathcal{A}$ knows (at least) one pair (m_1, c_1) encrypted with k

   $\mathcal{A}$ finds the key k, then decrypts any c: $k = m_1 \oplus c_1$, then $m_2 = k \oplus c_2$

3. *Chosen-plaintext attack (CPA):* $\mathcal{A}$ can obtain the encryption of a plaintext of his/her choice

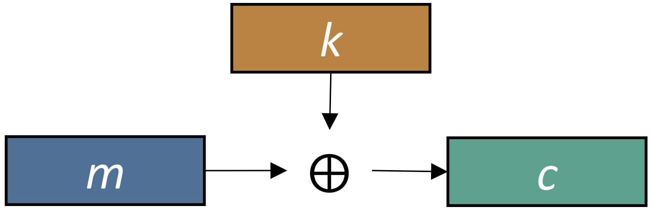
4. *Chosen-ciphertext attack (CCA):* $\mathcal{A}$ can obtain the decryption of a cipertext of his/her choice

   For 3 and 4, $\mathcal{A}$ can apply the same attack from 2.

One Time Pad (OTP)

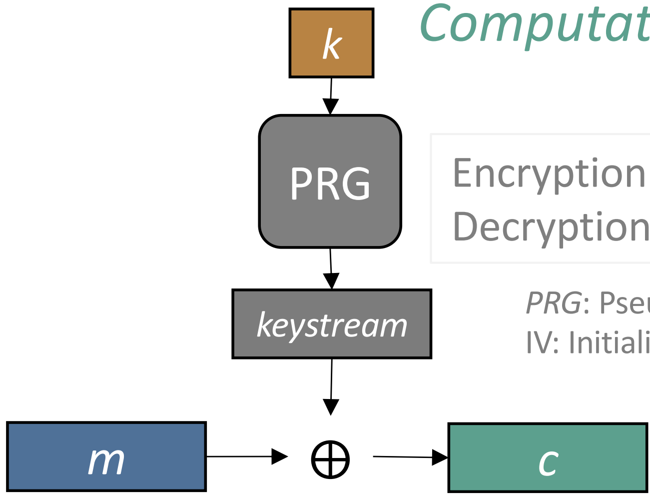
Perfect secrecy

Encryption: $c = k \oplus m$
Decryption: $m = k \oplus c$



Stream Ciphers

Computational secrecy

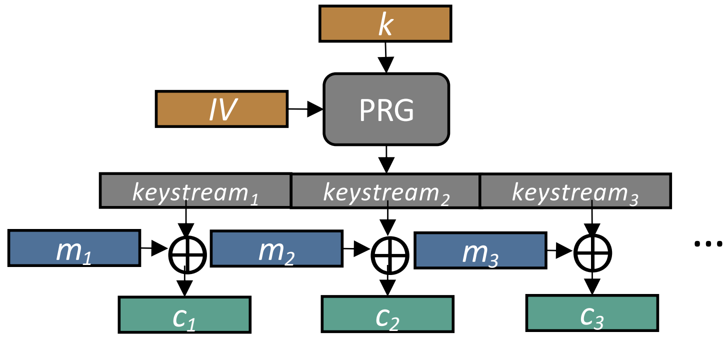


Encryption: $c = PRG(k) \oplus m$
Decryption: $m = PRG(k) \oplus c$

PRG: Pseudo-Random Generator
IV: Initialization Vector

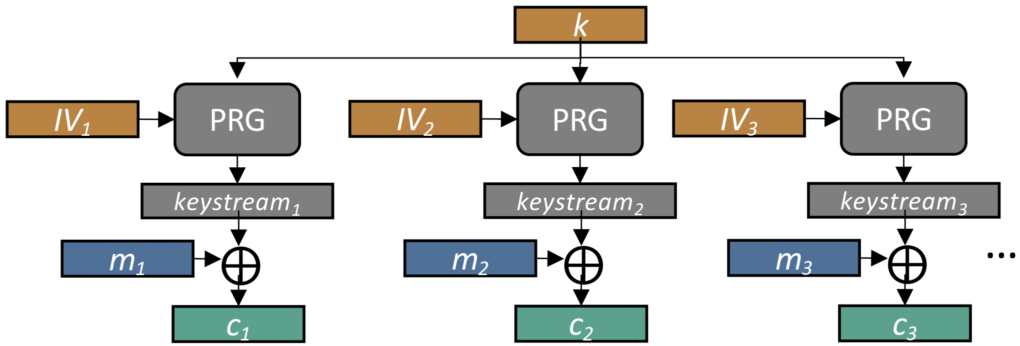
Synchronized Mode

Encryption: $c_1 \,||\, c_2 \,||\, c_3 \ldots = (IV, PRG(k,IV) \oplus m_1 \,||\, m_2 \,||\, m_3 \ldots)$
Decryption: $m_1 \,||\, m_2 \,||\, m_3 \ldots = PRG(k,IV) \oplus c_1 \,||\, c_2 \,||\, c_3 \ldots$
IV chosen uniformly at random



Unsynchronized Mode

Encryption: $c_i = (IV_i, PRG(k,IV_i) \oplus m_i)$
Decryption: $m_i = PRG(k,IV_i) \oplus c_i$
IV_1, IV_2, … chosen uniformly at random
(and thus independent)



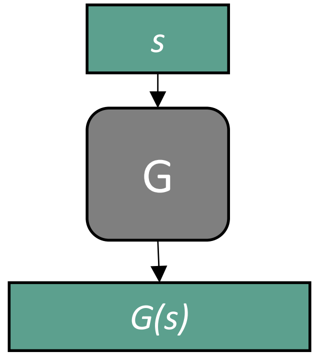
Pseudo-Random Generator (PRG)

G **deterministic** is *PRG* if for all *seed* with $|seed| = n$:

1. $l(n) = |G(s)| > |s| = n$ *(expansion)*
2. $\forall \mathcal{D}$ PPT, $\exists\ \varepsilon(n)$ negligible such that

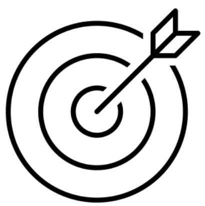
$$\mathrm{Adv}^{PRG}_{\mathcal{D},G}(n) = |\Pr[\mathcal{D}(r) = 1] - \Pr[\mathcal{D}(G(s)) = 1]| \leq \varepsilon(n)$$

where $r \leftarrow^{R} \{0,1\}^{l(n)}$ and $s \leftarrow^{R} \{0,1\}^{n}$ *(pseudo-randomness)*



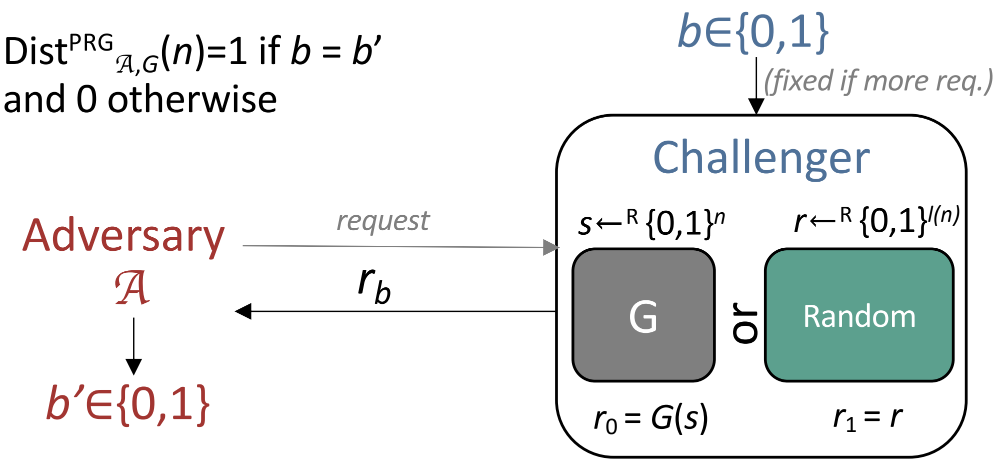
$\mathcal{D}$: distinguisher
$\mathcal{D}()$ output: 0 = not random, 1 = random
PPT: Probabilistic Polynomial in Time
$r \leftarrow^{R} \{0,1\}^{l(n)}$: r is random on $l(n)$ bits
$s \leftarrow^{R} \{0,1\}^{n}$: s is random on n bits

A **unpredictable** PRG is secure *(Theorem Yao'82)*
A **predictable** PRG is insecure!



Indistinguishability from random

$\mathrm{Dist}^{PRG}_{\mathcal{A},G}(n) = 1$ if $b = b'$
and 0 otherwise



G is PRG (cryptographically strong) if $\forall \mathcal{A}$ PPT,
$\exists\ \varepsilon(n)$ negligible such that:

$$\Pr[\mathrm{Dist}^{PRG}_{\mathcal{A},G}(n) = 1] \leq \frac{1}{2} + \varepsilon(n)$$

Semantic security

$b \in \{0,1\}$

Adversary $\mathcal{A}$ $\xrightarrow{m_0, m_1 \in \mathcal{M},\ |m_0| = |m_1|}$ **Challenger** $k \leftarrow \mathrm{Gen}(1^n)$

$c = \mathrm{Enc}(k, m_b)$

$b' \in \{0,1\}$

$\pi = (\mathrm{Enc}, \mathrm{Dec})$ is semantically secure if $\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible such that

$$\Pr[\mathrm{Priv}^{eav}_{\mathcal{A},\pi}(n) = 1] \le ½ + \varepsilon(n)$$

$\mathrm{Priv}^{eav}_{\mathcal{A},\pi}(n) = 1$ if $b' = b$ and 0 otherwise

CPA-security (Chosen-Plaintext Attack)

Semantic security + Adversary $\mathcal{A}$ $\xrightarrow{m'}$ **Encryption Oracle** $k(\leftarrow \mathrm{Gen}(1^n))$

$c' = \mathrm{Enc}(k, m')$

$\pi = (\mathrm{Enc}, \mathrm{Dec})$ is CPA-secure if $\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible such that

$$\Pr[\mathrm{Priv}^{cpa}_{\mathcal{A},\pi}(n) = 1] \le ½ + \varepsilon(n)$$

$\mathrm{Priv}^{cpa}_{\mathcal{A},\pi}(n) = 1$ if $b' = b$ and 0 otherwise

CCA-security (Chosen-Ciphertext Attack)

CPA-security + Adversary $\mathcal{A}$ $\xrightarrow{c' \neq c}$ **Decryption Oracle** $k(\leftarrow \mathrm{Gen}(1^n))$

$m' = \mathrm{Dec}(k, c')$

$\pi = (\mathrm{Enc}, \mathrm{Dec})$ is CCA-secure if $\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible such that

$$\Pr[\mathrm{Priv}^{cca}_{\mathcal{A},\pi}(n) = 1] \le ½ + \varepsilon(n)$$

$\mathrm{Priv}^{cca}_{\mathcal{A},\pi}(n) = 1$ if $b' = b$ and 0 otherwise

increased capabilities

Stronger security

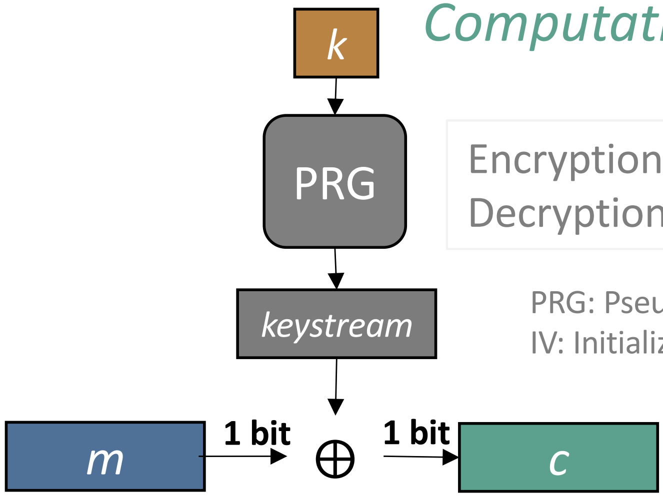
+ multiple interceptions / integrations

+ adaptive adversary

Π semantic secure at multiple interceptions ⇨ Π non-deterministic ; Π CCA-secure ⇨ Π non-malleable

Stream Ciphers

Computational secrecy



Encryption: $c = PRG(k) \oplus m$
Decryption: $m = PRG(k) \oplus c$

PRG: Pseudo-Random Generator
IV: Initialization Vector

- encrypts **bit-by-bit**
- the encryption of one bit is **independent** on the value of other bits in the plaintext (but only depends on the corresponding bit)

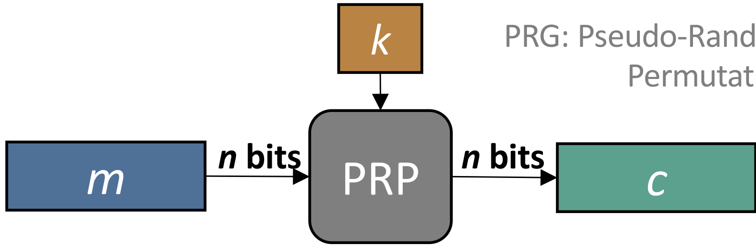
Less resources ⊕

Many broken ⊖

Block Ciphers

Computational secrecy

Encryption: $c = PRP(k,m)$
Decryption: $m = PRP^{-1}(k,c)$

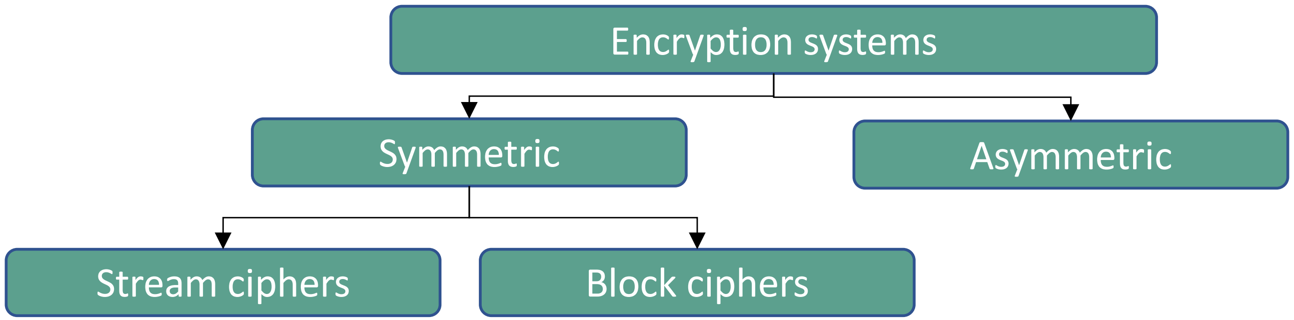


PRG: Pseudo-Random Permutation

- encrypts **in blocks of bits**
- the encryption is **dependent** on the values of all bits in plaintext block

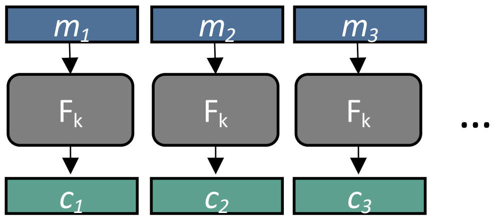
⊖ More resources

⊕ Seem more secure



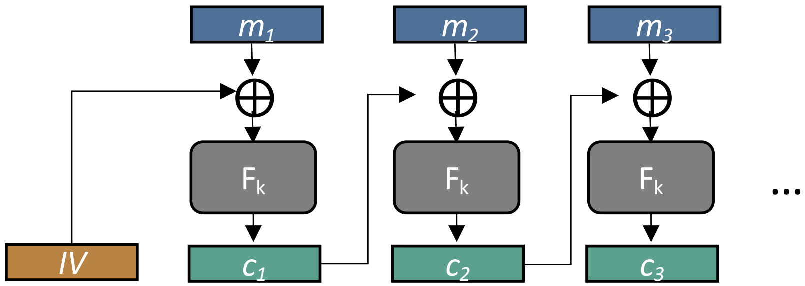
Modes of Operation (Block Ciphers)-

Electronic Code Book (ECB)



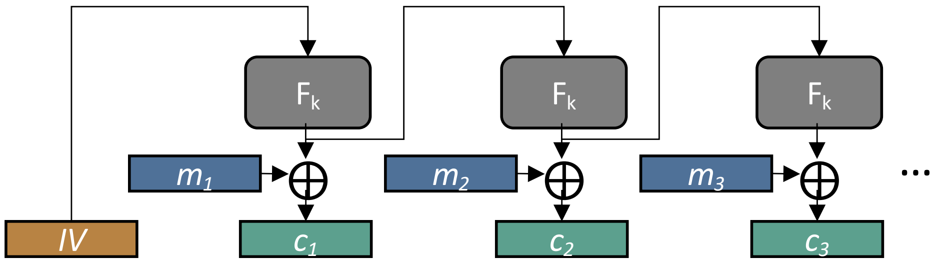
Encryption: $c_i = F(k,m_i)$
Decryption: $m_i = F^{-1}(k,c_i)$

Cipher Block Chaining (CBC)



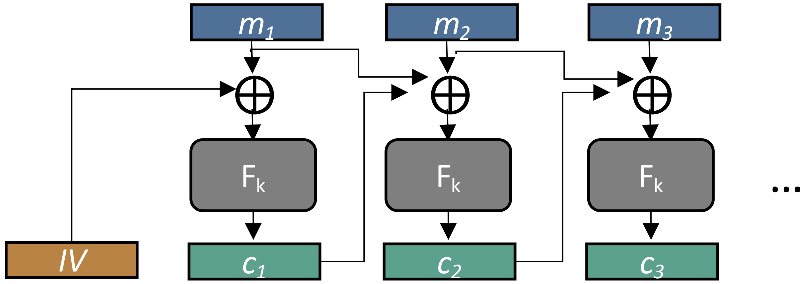
Encryption: $c_0 = IV$; $c_i = F(k,c_{i-1} \oplus m_i)$
Decryption: $m_i = F^{-1}(k,c_i) \oplus c_{i-1}$

Output Feedback (OFB)



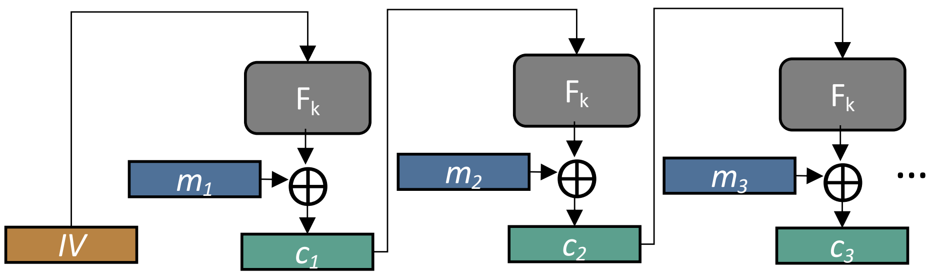
Encryption: $c_0 = IV$; $c_i = F^{(i)}(k,IV) \oplus m_i$
Decryption: $m_i = F^{(i)}(k,IV) \oplus c_i$

Propagating Cipher Block Chaining (PCBC)



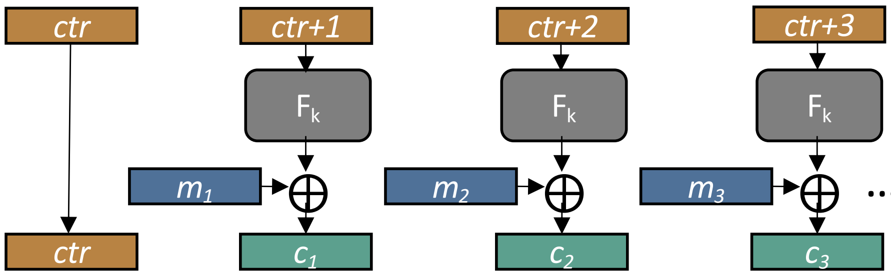
Encryption: $c_0 = IV$; $c_i = F(k,c_{i-1} \oplus m_{i-1} \oplus m_i)$
Decryption: $m_0 = 0^n$; $m_i = F^{-1}(k,c_i) \oplus c_{i-1} \oplus m_{i-1}$

Cipher Feedback Mode (CFB)



Encryption: $c_0 = IV$; $c_i = F(k, c_{i-1}) \oplus m_i$
Decryption: $m_i = F(k, c_{i-1}) \oplus c_i$

Counter Mode (CTR)



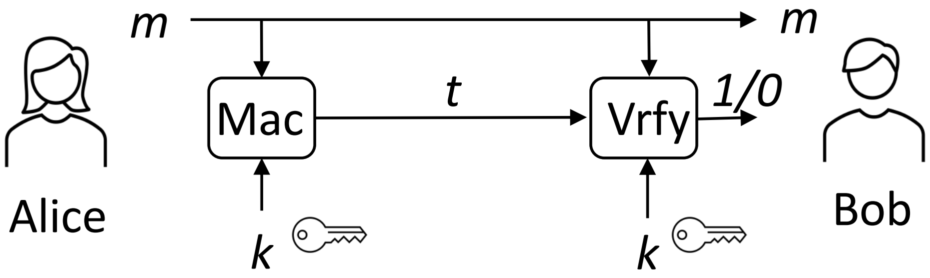
Encryption: $c_0 = ctr$; $c_i = F(k,ctr+i) \oplus m_i$
Decryption: $m_i = F(k,ctr+i) \oplus c_i$

*For simplicity in drawing, let F_k denote the PRF F with the key k given as input

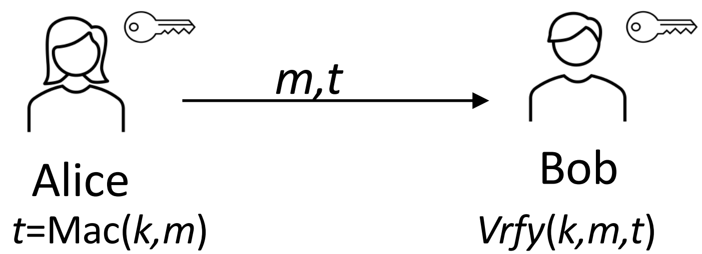
www.ruxandraolimid.weebly.com/pagesonsecurity

Pages on SecuRity
by Ruxandra F. Olimid

Message Authentication Code (MAC)-

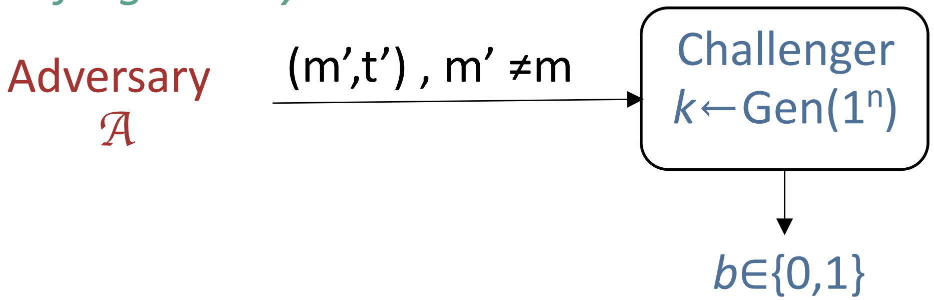


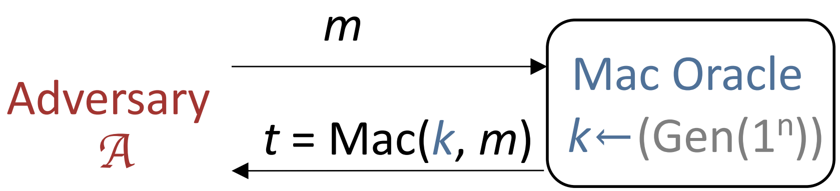
Tag generation: t = Mac(k,m)
Tag verification: $Vrfy(k,m,t)$ = 1 for a valid tag, 0 otherwise
Correctness: $\forall m \in \mathcal{M}$, $k \in \mathcal{K}$ Vrfy(k,m,Mac(k,m)) = 1



No. of keys	
for N bi-directional communicating parties	
Each: N-1 [k]	*Total: N(N-1)/2 [k]*

Unforgeability





*poly no. of queries

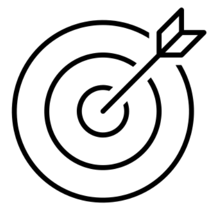
$\text{Mac}^{\text{forge}}_{\mathcal{A},\pi}(n)$ = 1 if Vrfy(k,m,t)=1 and 0 otherwise

$\pi = (\text{Mac}, \text{Vrfy})$ is existentially unforgeable under an adaptive chosen message attack if $\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible such that

$$\Pr[\text{Mac}^{\text{forge}}_{\mathcal{A},\pi}(n)=1] \le \varepsilon(n)$$

** Message Integrity Codes (MIC)*

Terminology		
k: symmetric key m: plaintext	t: tag	Mac: tag generation algorithm Vrfy: tag verification algorithm



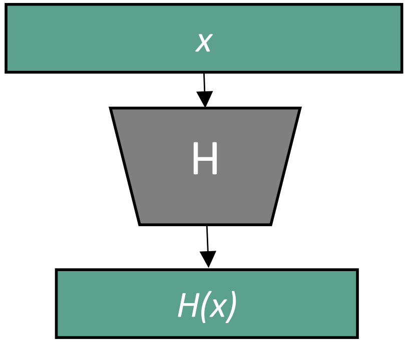
Integrity

Pages on SecuRity by Ruxandra F. Olimid

Hash Function -

Hash Function

$H: \{0,1\}^* \rightarrow \{0,1\}^{l(n)}$ *(fixed output length)*



l(n) = poly(n), with n the security parameter
$\{0,1\}^*$: sequence on bits, regardless its size
s.t.: such that
$\mathcal{A}$: adversary

Attacks:
Birthday attack

Security

Collision resistance

$\text{Hash}^{\text{coll}}_{\mathcal{A},H}(n)=1$ if:
$\mathcal{A}$ outputs
$x,y \in \{0,1\}^*$ s.t.
$\boldsymbol{x \neq y}$ **and** $\boldsymbol{H(x) = H(y)}$
$\text{Hash}^{\text{coll}}_{\mathcal{A},H}(n)=0$, otherwise

H is *collision resistant* if $\forall\mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible s.t.:
$\Pr[\text{Hash}^{\text{coll}}_{\mathcal{A},H}(n)=1] \leq \varepsilon(n)$

Second pre-image resistance

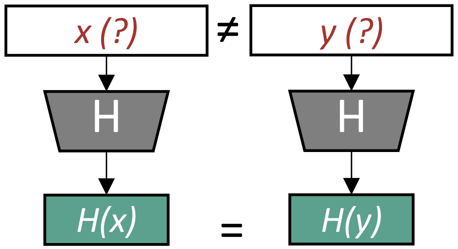
$\text{Hash}^{\text{2nd-pre-img}}_{\mathcal{A},H}(n)=1$ if:
given x $\in$ **{0,1}***, $\mathcal{A}$ outputs
$y \in \{0,1\}^*$ s.t.
$\boldsymbol{x \neq y}$ **and** $\boldsymbol{H(x) = H(y)}$
$\text{Hash}^{\text{2nd-pre-img}}_{\mathcal{A},H}(n)=0$, otherwise

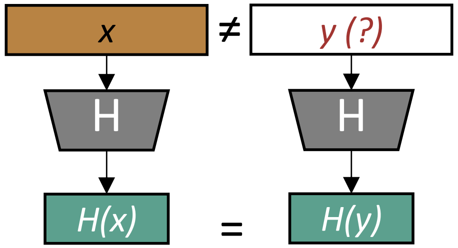
H is *second pre-image resistant* if $\forall\mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible s.t.:
$\Pr[\text{Hash}^{\text{2nd-pre-img}}_{\mathcal{A},H}(n)=1] \leq \varepsilon(n)$

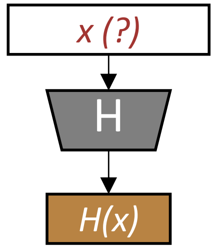
First pre-image resistance

$\text{Hash}^{\text{1st-pre-img}}_{\mathcal{A},H}(n)=1$ if:
given X, $\mathcal{A}$ outputs
$x \in \{0,1\}^*$ s.t.
$\boldsymbol{H(x) = X}$
$\text{Hash}^{\text{1st-pre-img}}_{\mathcal{A},H}(n)=0$, otherwise

H is *first pre-image resistant* if $\forall\mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible s.t.:
$\Pr[\text{Hash}^{\text{1st-pre-img}}_{\mathcal{A},H}(n)=1] \leq \varepsilon(n)$







one-way function

higher security ← *lower security*

Pages on SecuRity
by Ruxandra F. Olimid

Pseudo-Random Function (PRF)

A function F: $\mathcal{K}\times\mathcal{X}\rightarrow\mathcal{Y}$ such that:

1. $\forall\, k \in \mathcal{K}, x \in \mathcal{X}$, $\exists$ a PPT algorithm that (efficiently) computes $F_k(x)$ *(efficiency)*
2. $\forall$ algorithm PPT $\mathcal{D}$, $\exists\ \varepsilon(n)$ negligible such that

$$\mathrm{Adv}^{PRF}_{\mathcal{D},F}(n) = |\Pr[\mathcal{D}(f) = 1] - \Pr[\mathcal{D}(F_k(.)) = 1]| \leq \varepsilon(n)$$

where $f \leftarrow^R \mathrm{Func}(\mathcal{X}, \mathcal{Y})$ and $k \leftarrow^R \mathcal{K}$ *(pseudo-randomness)*

Pseudo-Random Permutation (PRP)

A bijection F: $\mathcal{X}\times\mathcal{K}\rightarrow\mathcal{X}$, with F PRF

$(\mathcal{Y} = \mathcal{X})$

$\mathcal{D}$: distinguisher

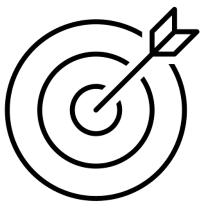
$\mathcal{D}()$ output: 0 = not random, 1 = random

PPT: Probabilistic Polynomial in Time

$\mathrm{Func}(\mathcal{X}, \mathcal{Y})$: the set of all functions from $\mathcal{X}$ to $\mathcal{Y}$, $\mathcal{K}=\{0,1\}^n$

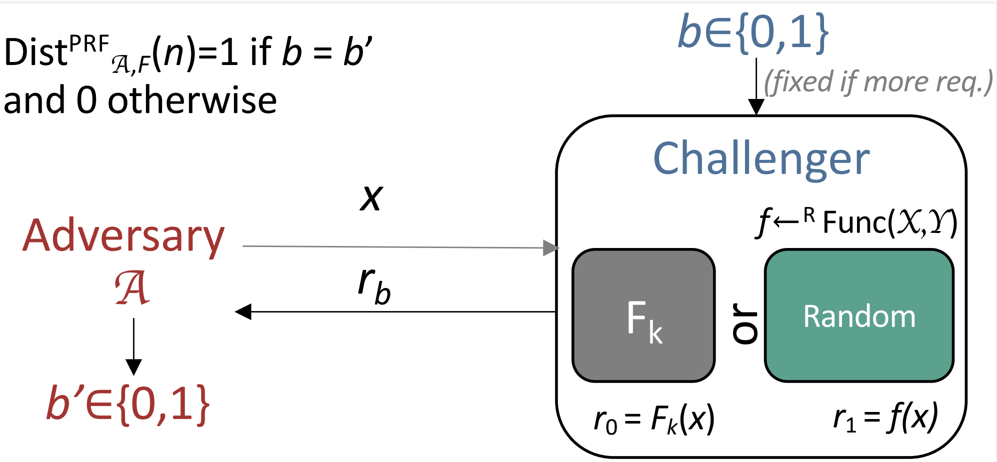
$f \leftarrow^R \mathrm{Func}(\mathcal{X}, \mathcal{Y})$: f is random function in $\mathrm{Func}(\mathcal{X}, \mathcal{Y})$

$k \leftarrow^R \mathcal{K}$: k is random key



Indistinguishability from random functions / permutations

$\mathrm{Dist}^{PRF}_{\mathcal{A},F}(n)=1$ if $b = b'$ and 0 otherwise



F is PRF if $\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible such that:

$$\Pr[\mathrm{Dist}^{PRF}_{\mathcal{A},F}(n)=1] \leq ½ + \varepsilon(n)$$

Diffie-Hellman Key Exchange -

www.ruxandraolimid.weebly.com/pagesonsecurity

DLP

- $\mathcal{A}$ is given: (G,q,g,A) with *G* cyclic group of order *q, g* generator and $A = g^a$, $a \leftarrow^R Z_q$
- $\mathcal{A}$ returns: a’ in Z_q

The experiment outputs:

1 if $A = g^{a'}$, 0 otherwise

$\forall\mathcal{A}$ PPT, $\exists\, \varepsilon(n)$ negligible such that:

$$\Pr[DLP_{\mathcal{A}}(n)=1] \le \varepsilon(n)$$

CDH

- $\mathcal{A}$ is given: (G,q,g,A,B) with *G* cyclic group of order *q, g* generator, $A = g^a, B = g^b$, $a,b \leftarrow^R Z_q$
- $\mathcal{A}$ returns: K in Z_q

The experiment outputs:

1 if $K = g^{ab}$, 0 otherwise

$\forall\mathcal{A}$ PPT, $\exists\, \varepsilon(n)$ negligible such that:

$$\Pr[CDH_{\mathcal{A}}(n)=1] \le \varepsilon(n)$$

DDH

$\forall\mathcal{A}$ PPT, $\exists\, \varepsilon(n)$ negligible such that:

$$\Pr[\mathcal{A}(G,q,g,g^a,g^b,g^c)=1] - \Pr[\mathcal{A}(G,q,g,g^a,g^b,g^{ab})=1] \le \varepsilon(n)$$

for $a,b,c \leftarrow^R Z_q$

Stronger security

DLP: Discrete Logarithm Problem
DDH: Decisional Diffie-Hellman Problem
CDH: Computational Diffie-Hellman Problem

Diffie-Hellman Key Exchange

Alice		Bob
	(G,q,g) ⟷	
$a \leftarrow^R Z_q$ $A=g^a$	A →	
	← B	$b \leftarrow^R Z_q$ $B=g^b$
$k_A=B^a$		$k_B=A^b$

Attacks: no authentication of parties, Man-in-the-Middle

Man-in-the-Middle

Alice				Bob
	(G,q,g) ⟷		(G,q,g) ⟷	
$a \leftarrow^R Z_q$ $A=g^a$	A →	$a', b' \leftarrow^R Z_q$ $B'=g^{b'}$ $A'=g^{a'}$		
	← B’		A’ →	
			← B	$b \leftarrow^R Z_q$ $B=g^b$
$k_A=B'^a$		$k_A=A^{b'}$ $k_B=B^{a'}$		$k_B=A'^b$

Pages on SecuRity
by Ruxandra F. Olimid