Introduction to cryptology (GBIN8U16)

✧

Passive encryption

Pierre Karpman
pierre.karpman@univ-grenoble-alpes.fr
https://membres-ljk.imag.fr/Pierre.Karpman/tea.html
https://membres-ljk.imag.fr/Bruno.Grenet/IntroCrypto.html

2024–02–07

Passive encryption; large shared secret

Passive encryption; small shared secret

The goal: confidentiality

Current context:

- Two persons $\mathscr{A}$ & $\mathscr{B}$ wish to communicate over a reliable channel
  - Wlog: one-way communication
  - Wlog: messages are always 128-bit long
- Passive adversaries

$\rightsquigarrow$ encryption scheme to be evaluated w.r.t. IND-CPA security

How to do it if $\mathscr{A}$ & $\mathscr{B}$:

- Know a large *shared secret*?
- — small — ?

Passive encryption; large shared secret

Passive encryption; small shared secret

Large shared secret

Assumptions:

- $\mathscr{A}$ & $\mathscr{B}$ can share a "large" K (e.g. $K \in \{0,1\}^{128 \times 2^{128}}$) a priori known only by themselves
  $\rightsquigarrow$ *Symmetric* / *secret-key* cryptography
- $\mathscr{A}$ can draw *uniform* and independent random bits at will
  - And so a uniform, arbitrarily long bitstring

Objective:

- Using those capabilities to build an encryption scheme with good IND-CPA security

A first encryption scheme

(An instance of) *one-time pad*, OTP128 :

1. $\mathscr{A}$ draws a uniform bitstring $K \in \{0,1\}^{128 \times 2^{128}}$ and shares it with $\mathscr{B}$
2. $\mathscr{A}$ sets a counter i to 0
3. Every time $\mathscr{A}$ wishes to send a message $m \in \{0,1\}^{128}$, he selects the bits K_i of K of indices $i \cdots i + 127$ and sends $(i, m \oplus K_i)$ to $\mathscr{B}$, and then increments i by 128. If $i = 2^{128}$, $\mathscr{A}$ cannot send messages with this system any more.

Remark: This encryption scheme is randomised: a unique plaintext may map to many ciphertexts (Q: how many?)

Same scheme, but with functions

1. $\mathscr{A}$ draws a function $\mathsf{K} \in \{0,1\}^{128} \to \{0,1\}^{128}$ uniformly from the set of all such functions, and shares it with $\mathscr{B}$
2. $\mathscr{A}$ sets a counter i to 0
3. Every time $\mathscr{A}$ wishes to send a message $m \in \{0,1\}^{128}$, he sends $(i, m \oplus \mathsf{K}(i))$ to $\mathscr{B}$, then increments i by 128. If $i = 2^{128}$, $\mathscr{A}$ cannot send messages with this system any more.

IND-CPA security of OTP

To analyse the IND-CPA security of OTP, we rely on the key lemma (and its generalisations):

Lemma ($\mathcal{U} \oplus * \approx \mathcal{U}$)

Let X be a uniform random variable over $\{0, 1\}$ and Y an independent random variable following an arbitrary distribution $\{0, 1\}$, then $Z := X \oplus Y$ is uniform and independent from Y.

Proof $\rightsquigarrow$ TD

Remark: This result and its (many) generalisations is essential in cryptography, and used in many constructions

IND-CPA security of OTP128

- We assume a deterministic adversary A (one may show that this is wlog) that makes $q < 2^{128}$ training queries
- Def. $\mathcal{O} := \{c_b : \mathsf{A}(\{(x_i, y_i)_{1 \le i \le q}\}, m_0, m_1, c_b) = 1\}$
- The success probability is measured over the sampling of b and K
- $\Pr[\hat{b} = b] = \Pr[\mathsf{A}(\cdots) = 0 \wedge b = 0] + \Pr[\mathsf{A}(\cdots) = 1 \wedge b = 1]$
  $= 1/2 \times (\Pr[\mathsf{A}(\cdots) = 0 : b = 0] + \Pr[\mathsf{A}(\cdots) = 1 : b = 1])$
- $p_1 := \Pr[\mathsf{A}(\cdots) = 1] = \Pr[c_b \in \mathcal{O}] = \#\mathcal{O}/2^{128}$
  $p_0 := \Pr[\mathsf{A}(\cdots) = 0] = \Pr[c_b \in \overline{\mathcal{O}}] = 1 - p_1$
- $\Pr[c_b \in \mathcal{O} : b = 0] = \Pr[c_b \in \mathcal{O} : b = 1] = p_0$
- $\Pr[c_b \in \overline{\mathcal{O}} : b = 0] = \Pr[c_b \in \overline{\mathcal{O}} : b = 1] = p_1$
- $\Pr[\hat{b} = b] = 1/2 \rightsquigarrow \mathbf{Adv}^{\text{IND-CPA}}_{\text{OTP128}}(< 2^{128}, \infty) = 0$

IND-CPA security of OTP128 (bis)

Remarks:

- $\mathbf{Adv}^{\text{IND-CPA}}_{\text{OTP128}}(< 2^{128}, \infty) = 0 \rightsquigarrow$ the best we could hope for: whatever the computational power of the adversary, its advantage is zero $\rightsquigarrow \infty$ bits of security (whatever the definition)
- Sometimes called "information theoretic" perfect security (zero advantage)
  - **WARNING**: we achieved this thanks to very strong assumptions on our capabilities and on the adversaries
- One also gets a zero advantage w.r.t. stronger variants of the IND-CPA definition
  - One may exactly *simulate* OTP128 without knowing the messages

Passive encryption; large shared secret

Passive encryption; small shared secret

Decreasing the secret size

- (An instance of) OTP provides the best (passive) security one could hope for
- But needs a large shared secret
- But ∞ security not needed; 128 bits of security would (often) be enough
- Objective: reduce the secret size, while keeping a good (not ∞) security level

Small shared secret

Assumptions:

- $\mathscr{A}$ & $\mathscr{B}$ can share a "small" K (e.g. $K \in \{0,1\}^{128}$) a priori known only by themselves
- $\mathscr{A}$ can draw *uniform* and independent random bits at will

Objective:

- Using those capabilities to build an encryption scheme with good IND-CPA security, *possibly under additional assumptions TBD*

Enter primitives!

Ideas:

- Adding uniform independent randomness gives infinite IND-CPA security
- But not enough randomness to do this for every message
- $\rightsquigarrow$ "stretch" our small uniform randomness to a large *almost uniform* one, and use the latter?
- $\rightsquigarrow$ use for this a good (family of) *pseudorandom function* (a *primitive*)
- $\rightsquigarrow$ if the function is "good" (in a **precise sense** (TBD)), then get "good" IND-CPA security

Pseudorandom function family: syntax

- Usually, one considers $\mathsf{F} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$: a family of functions in one parameter (usually called the *key*): for all $k \in \{0,1\}^\kappa$, $\mathsf{F}(k, \cdot)$ is a function $\{0,1\}^n \to \{0,1\}^n$
- (Sometimes, one rather wants functions with variable-size input/output)

Encryption from pseudorandom functions

1. $\mathscr{A}$ and $\mathscr{B}$ *publicly* agree on a family of functions $\mathsf{F} : \{0,1\}^{128} \times \{0,1\}^{128} \to \{0,1\}^{128}$
2. $\mathscr{A}$ draws a uniform $K \in \{0,1\}^{128}$ and shares it with $\mathscr{B}$
3. $\mathscr{A}$ sets a counter i to 0
4. Every time $\mathscr{A}$ wishes to send a message $m \in \{0,1\}^{128}$, he sends $(i, m \oplus \mathsf{F}(K,i))$ to $\mathscr{B}$, then increments i by 128. If $i = 2^{128}$, $\mathscr{A}$ cannot send messages with this system any more.

$\rightsquigarrow$ an instance of the *counter mode* (CTR) for the (family of functions) F (notation: CTR[F])

Pseudorandom functions: security (with CTR encryption in mind)

Ideas:

- The only difference between OTP128 = CTR[K] and CTR[F] is swapping $\mathsf{K} \twoheadleftarrow \mathsf{Funcs}(\{0,1\}^{128})$ for $\mathsf{F}(K,\cdot), K \twoheadleftarrow \{0,1\}^{128}$
  - Notation: For a finite set $\mathcal{S}$, $\mathsf{Funcs}(\mathcal{S})$ denotes the set of functions $\mathcal{S} \to \mathcal{S}$
  - Notation: For a finite set $\mathcal{S}$, $X \twoheadleftarrow \mathcal{S}$ denotes the fact that X is drawn uniformly from $\mathcal{S}$, independently from other drawings
- If one assumes that it's hard for every adversary to distinguish K from a uniform member of F, then it will be hard to distinguish CTR[F] from OTP128, which has infinite security
  - *Reduction proof*: reduce the IND-CPA security of CTR[F] to the *PRF* security of F

PRF security

One defines PRF (*pseudorandom function*) security of a family of function F through the advantage function:

$\mathbf{Adv}^{\mathsf{PRF}}$

$\mathbf{Adv}_{\mathsf{F}}^{\mathsf{PRF}}(q,t) =$

$$\max_{A_{q,t}} \Big| \Pr[A_{q,t}^{\mathbb{O}}() = 1 : \mathbb{O} \twoheadleftarrow \mathsf{Funcs}(\{0,1\}^n)] - \Pr[A_{q,t}^{\mathbb{O}}() = 1 : \mathbb{O} = \mathsf{F}(K,\cdot), K \twoheadleftarrow \{0,1\}^{\kappa}] \Big|$$

Remark: Abusing terminology, one often says that F **is** a PRF to mean that it has "good" PRF security. (Same thing for "Enc is an IND-CPA encryption scheme")

PRF security: remarks (bis)

- For all $\kappa \lessapprox n2^n$, for all F, one may win $\mathbf{Adv}_\mathsf{F}^\mathsf{PRF}$ with constant advantage given sufficiently-many resources q and t
- For instance for $\kappa = n$, $\mathbf{Adv}_\mathsf{F}^\mathsf{PRF}(2, 2^n) \approx 1$ (Cf. TD)
- Those are *generic attacks*: only "the parameters" are attacked; not the functions specifically

$\rightsquigarrow$ One must pay **ATTENTION** to the parameter size, so that they resist generic attacks (cf. OOM of computations/advantages)

Encryption from pseudorandom functions: security

- One may show that for $q < 2^n$, $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[F]}}(q, t) \approx \mathbf{Adv}^{\text{PRF}}_{\text{F}}(q, t)$ (for our "one-block" messages)
- $\rightsquigarrow$ a good PRF is enough to get good IND-CPA encryption
- "any" good PRF $\rightsquigarrow$ modularity
- More generally, encryption schemes are (very) often built as *mode of operation* on top of functions, *block ciphers* (cf. below) etc.

Pseudorandom functions: construction

- It is in fact easier to build families of pseudorandom *permutations* than (arbitrary) functions $\rightsquigarrow$ *block cipher*

Definition: permutation

A permutation is a bijective function from a finite set to itself. There are $N!$ distinct permutations over a set of N elements.

Definition: block cipher

A block cipher is a family of permutations: a function $\mathsf{E} : \{0,1\}^\kappa \times \mathcal{M} \to \mathcal{M}$ s.t. $\forall k \in \{0,1\}^\kappa$ $\mathsf{E}(k,\cdot)$ is a permutation

Remark: In general, $\mathcal{M} = \{0,1\}^n$ for $n \in \{64, 128, 256\}$

PRP security

One defines the PRP (*pseudorandom permutation*) security of a block cipher E with messages in $\{0,1\}^n$ through the advantage function:

Adv$^{\mathsf{PRP}}$

$\mathbf{Adv}_{\mathsf{E}}^{\mathsf{PRP}}(q,t) =$

$$\max_{A_{q,t}} \Big| \Pr[A_{q,t}^{\mathbb{O}}() = 1 : \mathbb{O} \twoheadleftarrow \mathsf{Perms}(\{0,1\}^n)] - \Pr[A_{q,t}^{\mathbb{O}}() = 1 : \mathbb{O} = \mathsf{E}(K,\cdot), K \twoheadleftarrow \{0,1\}^\kappa] \Big|$$

- $\mathsf{Perms}(\mathcal{S})$: the set of all permutations over $\mathcal{S}$

PRP/PRF switching

- Swapping a PRF for a PRP (e.g. in CTR mode) only (provably) preserves security if a good PRP is also a good PRF. So:

Lemma (PRP/PRF switching)

Let E be a family of permutations over N elements, one has:

$$\mathbf{Adv}_{\mathsf{E}}^{\mathsf{PRF}}(q,t) \leq \mathbf{Adv}_{\mathsf{E}}^{\mathsf{PRP}}(q,t) + \frac{q(q-1)}{2N}$$

Remarks:

- The term $q(q-1)/2N$ is *generic* (it does not depend on E)
- It is a "birthday" term (cf. the "birthday paradox") and the inequality becomes vacuous at the "birthday bound" i.e. when $q \approx \sqrt{N}$
- This bound is *tight* (for $q \leq \sqrt{2N}$, $t \propto q$, lower-bounded by $q(q-1)/4N$)

Security of the CTR mode with block ciphers

- From the above:

$$\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(q,t) \approx \mathbf{Adv}^{\text{PRF}}_{\text{E}}(q,t) \lessapprox \mathbf{Adv}^{\text{PRP}}_{\text{E}}(q,t) + \frac{q(q-1)}{2^{n+1}}$$

- $\rightsquigarrow$ (Any) good PRP is enough to build a good IND-CPA encryption scheme
- One also gets a lower-bound (cf. supra): security collapses at the birthday bound
- $\rightsquigarrow$ The (IND-CPA) of CTR mode depends on the (PRP) security of the block cipher **BUT ALSO** on the volume of encrypted data
  - $\rightsquigarrow$ One must stop communications/change key *well before* $q \approx \sqrt{2^n} = 2^{n/2}$

Birthday-bound security: impact

Numerical application:

- E with 64-bit blocks
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{10}, 2^{10}) \gtrapprox 2^{-46}$ (64 Kb encrypted data)
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{20}, 2^{20}) \gtrapprox 2^{-26}$ (64 Mb encrypted data)
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{30}, 2^{30}) \gtrapprox 2^{-6}$ (64 Gb encrypted data)
- — 128 bits
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{30}, 2^{30}) \gtrapprox 2^{-70}$ (128 Gb encrypted data)
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{60}, 2^{60}) \gtrapprox 2^{-10}$ (128 Eb encrypted data)
- — 256 bits
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{60}, 2^{60}) \gtrapprox 2^{-138}$ (128 Eb encrypted data)
  - $\mathbf{Adv}^{\text{IND-CPA}}_{\text{CTR[E]}}(2^{80}, 2^{80}) \gtrapprox 2^{-98}$ (128 Yb encrypted data)

$\rightsquigarrow$ No worries with large blocks, but careful with small ones!!

$\rightsquigarrow$ Can lead to real-life attacks, e.g. `https://sweet32.info/`

So now that we have block ciphers...

Mode of operation (informally)

A *(block cipher) mode of operation* for encryption is an algorithm that builds an encryption scheme:

$$\mathsf{Enc} : \{0,1\}^\kappa \times \cdots \times \{0,1\}^* \to \{0,1\}^*$$

from a block cipher:

$$\mathsf{E} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$$

- CTR is *one* “good” mode: the result “is” IND-CPA if the block cipher “is” a PRP
- What else can we do... Can alternatives give us better (*beyond the birthday bound* (BBB)) security?

An easy alternative (that turns out to be stupid): ECB

Electronic CodeBook: just concatenate independent calls to E

Electronic Code Book mode

$\mathsf{Enc}(k, m_0 || m_1 || \ldots) \mapsto \mathsf{E}(k, m_0) || \mathsf{E}(k, m_1) || \ldots$

- No security
  - Exercise: give a simple attack on ECB for the IND-CPA security notion w/ advantage 1, low complexity

Another alternative (this time it's good): CBC

Cipher Block Chaining: Chain blocks together (duh)

Cipher Block Chaining mode

$\mathsf{Enc}(k, r, m_1||m_2||\ldots) \mapsto$
$c_0 := r||c_1 := \mathsf{E}(k, m_1 \oplus c_0)||c2 := \mathsf{E}(k, m_2 \oplus c_1)||\ldots$

- Output block i (ciphertext) added (XORed) to input block $i+1$ (plaintext)
- For first (m_0) block: use "random" IV r ($\leftarrow$ one more parameter to Enc (or not... depends how we see it))
- (Q: how do you decrypt (assuming you know k, E^{-1}?))
- What security?

CBC IVs

CBC has bad IND-CPA security if the IVs are not unpredictable by the adversary

- Consider an IND-CPA adversary that asks an oracle query $\mathrm{CBC}[\mathrm{E}](m)$, gets $r, c = \mathrm{E}(k, m \oplus r)$
- Assume the adversary knows that for the next IV r', $\Pr[r' = x]$ is large
- Sends two challenges $m_0 = m \oplus r \oplus x$, $m_1 = m_0 \oplus 1$
- Gets $c_b = \mathrm{CBC}(m_b)$, $b \twoheadleftarrow \{0, 1\}$
- If $c_b = c$, guess $b = 0$, else $b = 1$

Generic CBC collision attack

Even with unpredictable IVs, CBC can be attacked

An observation:

- For a fixed k, $\mathrm{E}(k, \cdot)$ is a permutation so $\mathrm{E}(k, x) = \mathrm{E}(k, y) \Leftrightarrow x = y$
- In CBC, inputs to E are of the form $x \oplus y$ where x is a message block and y an IV or a ciphertext block
- So $\mathrm{E}(k, x \oplus y) = \mathrm{E}(k, x' \oplus y') \Leftrightarrow x \oplus y = x' \oplus y'$

A consequence:

- If $c_i = \mathrm{E}(k, m_i \oplus c_{i-1}) = c'_j = \mathrm{E}(k, m'_j \oplus c'_{j-1})$, then $m_i \oplus c_{i-1} = m'_j \oplus c'_{j-1}$, and then $c_{i-1} \oplus c'_{j-1} = m_i \oplus m'_j$
- $\rightsquigarrow$ knowing identical ciphertext blocks reveals information about the message blocks (Or IV... then no worries (but unlikely))
- $\Rightarrow$ breaks IND-CPA security (regardless of how good (e.g. of a PRP) E is)

CBC collisions: how likely?

How soon does a collision happen?

- Assumption: the distribution of the $(x \oplus y)$ is $\approx$ uniform
  - If y is an IV it has to be (close to) uniformly random, otherwise we have an attack (two slides ago)
  - If $y = \mathrm{E}(k, z)$ is a ciphertext block, ditto for y knowing z, otherwise we have a PRP attack on E
- $\Rightarrow$ A collision occurs w/ prob. $\approx q^2/2^n, q \leq 2^{n/2}$ for q the *total* number of calls to E across (possibly) multiple messages $\leftarrow$ the birthday bound again! (So CBC not BBB)
- One may show that this attack is essentially optimal (w/o exploiting possible weaknesses of E)

$\rightsquigarrow$

$$\mathbf{Adv}^{\text{IND-CPA}}_{\text{CBC[E]}}(q, t) \lessapprox \mathbf{Adv}^{\text{PRP}}_{\text{E}}(q, t) + \frac{q^2}{2^n}$$

Another alternative (this time it's BBB): CENC

CENC: the (basic) idea:

- CTR mode with a (raw) PRP is not BBB because a (raw) PRP is not a BBB PRF
- But if one could build a BBB PRF from a PRP, it would be enough to use this PRF in CTR mode to get BBB security!
- $\mathsf{XP[E]}(k, x) \mapsto \mathsf{E}(k, 0||x) \oplus \mathsf{E}(k, 1||x)$ is a BBB $(n-1)$-bit PRF construction from an n-bit PRF:

$$\mathbf{Adv}^{\mathsf{PRF}}_{\mathsf{XP[E]}}(q, t) \lessapprox \mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{XP[E]}}(q, t) + \frac{q}{2^n}$$

- But using XP[E] instead of E is $\approx$ twice more expensive! So CENC trades (a bit of) efficiency for (a bit of) security (while remaining BBB) $\rightsquigarrow$ for more details, cf. Iwata 2006

Block ciphers, functions... other definitions

Many usage of block ciphers/function reduce to PRP/PRF security, but there are alternatives, e.g.:

- Ideal (non-standard) models (cf. next lecture)
- *search-based* (rather than *decision-based* definitions, e.g. *unpredictability*: typically not appropriate for encryption, but appropriate for authentication (cf. —)
- PRP/PRF with related keys, key-dependent messages etc.

Unpredictability

To attack the *unpredictability* of a BC $\mathsf{E} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$, define:

Game $\mathsf{Forge}^{\mathsf{E}}$

Give the adversary oracle access to $\mathbb{O} = \mathsf{E}(k, \cdot)$ for $k \twoheadleftarrow \{0,1\}^\kappa$
The adversary wins iff. it returns a couple (x, y) s.t.:

1. x was not queried to $\mathbb{O}$
2. $\mathsf{E}(k, x) = y$

$\rightsquigarrow$

$\mathbf{InSec}^{\mathsf{UP}}$

$\mathbf{InSec}^{\mathsf{UP}}_{\mathsf{E}}(q, t) = \max_{A_{q,t}} \Pr[A^{\mathbb{O}}_{q,t}() \text{ wins } \mathsf{Forge}^{\mathsf{E}}]$

Where $A_{q,t}$ run in time t and make q queries to its oracle

A few examples of block ciphers

- AES ("Advanced Encryption Standard"): 128-bit blocks; 128, 192 or 256-bit keys
  - NIST Standard (USA): FIPS 197 (2001)
  - Versatile, good performance, studied a lot, no known vulnerability (when used in a "normal" context)
- PRESENT: 64-bit blocks; 80 or 128-bit keys
  - An example of *lightweight block cipher*: cheap on ASICs
- SPECK: 48 to 128-bit blocks; 96 to 128-bit keys
  - Another lightweight block cipher: cheap in software (on small CPUs). Mind the very small blocks!
- SHACAL-2: 256-bit blocks; 512-bit keys
  - An example of block cipher with large blocks and a very large key

Conclusion (so far)

- (IND-CPA) encryption schemes from (PRP) block ciphers in mode of operation: a common approach but not the only one!
  - Example of alternative: permutation-based encryption, e.g. ASCON (in the process of being standardised by NIST)
- Security definitions for functions and block ciphers: PRF, PRP... and others!
  - Other definitions and models exist, e.g. *unpredictability* (UP), ideal models (cf. OTP, function view; *ideal block ciphers* (in a next lecture))
- Remember: IND-CPA security only considers weak *passive* adversaries
  - But an IND-CPA encryption scheme is a good starting point to eventually get something that also provides confidentiality v. active adversaries! (cf. next)

Conclusion (so far, bis)

In practice, symmetric encryption is (very) efficient. Some orders of magnitude (using appropriate algorithms):

- On a high-end architecture: only a few CPU cycles to encrypt one byte
- On a low-end architecture: only a few hundred bytes to implement encryption and a few dozen cycles to encrypt one byte (Warning: w/o protection against side channels!)
- On a circuit: only a few thousand gates to implement encryption (Warning: w/o protection against side channels!)