Admin: * Pset #1 posted, due Feb 26.
  * Announcement: Daniel Genkin giving a talk at crypto day, Friday Feb 16, 2:45-3:45, 6-882 on Spectre & Meltdown.

Today:  - Enc
  - One time pad. (OTP)

Encryption

Goal: Confidentiality of transmitted (or stored) msg.

Parties: Alice & Bob are "good guys"
  Eve is "eavesdropper" / "adversary"

Alice $\to$ Eve $\to$ Bob

transmitted msg

* Eve can see all the msgs sent on the channel, but should not learn $M$.

In basic picture above, there is nothing to distinguish Eve from Bob.

How do we ensure that Bob receives $M$, but
Eve does not?

**Crypto Approach**

- Bob knows a **key** $K$ that Eve doesn’t (Eve knows the system, but not the key $K$).
- Alice can **encrypt** $M$ so that knowledge of $K$ allows for decryption.
- Eve sees ciphertext, but learns nothing about $M$.

**Classical (non public-key) crypto:**

Alice & Bob both know key $K$.
($K$ is shared symmetric key)

**Algorithms**:

\[ K \leftarrow \text{Gen}(1^n) \]: Generates key of length $n$.
($n$ is given to Gen in unary, so we can say that Gen is a (probabilistic) poly time algorithm).

\[ C \leftarrow \text{Enc}(K, M) \]: Encrypts msg $M$ with key $K$. Result is $C$.

\[ M = \text{Dec}(K, C) \]: Decrypts $C$ using $K$ to obtain $M$. 
Convention:
"\leftarrow" for randomized computations
(often \(\leftarrow^R\) or \(\leftarrow^\$\) is used).

"=\) for deterministic computations.

Correctness Requirement: \(\forall M\)

\[\Pr[\text{Dec}(K, \text{Enc}(K, M)) = M] = 1 - \text{negl}(\lambda)\]

Where the prob. is over \(K \leftarrow \text{Gen}(\lambda^3)\) and over the randomness of Enc & Dec.

\(\text{negl}(\lambda)\): A function that approaches 0 faster than \(\frac{1}{\text{poly}(\lambda)}\) (for every poly).

(Formally: \(\mu(\lambda) = \text{negl}(\lambda)\) if \(\forall \text{ const. } c \in \mathbb{N} \in \text{ NE} \exists \text{ poly } s.t.

\(\forall N > N\r
\mu(\lambda) < \frac{1}{N^c}\)

In crypto we are willing to tolerate negl error

* Why are Gen & Enc probabilistic? For security.

Why is Dec deterministic? Randomness is not needed for decryption.
Setup: Someone computes $K \leftarrow \text{Gen}(1^n)$
(may be Alice or Bob)

and ensures that both Alice & Bob have $K$
(and Eve doesn't). (How ??)

Alice($K$) \[ C \]

Bob($K$)

$C \leftarrow \text{Enc}(K, M)$

Eve ??

Security Definition:

Objective: Eve cannot distinguish $\text{Enc}(K, M_1)$ from $\text{Enc}(K, M_2)$ even if she knows (or chooses) $M_1, M_2$
(of same length)
[Encryption typically does not hide msg length]

This security notion is called "semantic security"
[Goldwasser - Micali 82].
(also called "ciphertext indistinguishability")

Formal Def: \[ \forall \text{ Eve} \quad \forall \text{ } \lambda \in \mathbb{N} \quad \forall M_0, M_1 \text{ adversarially chosen by Eve (given } 1^n) \]

\[ \Pr \left[ \text{ Eve } \left( 1^n, \text{ Enc}(K, M_b) \right) = b \right] = \frac{1}{2} + \text{negl}(\lambda) \]
This is known as a “game-based” definition.

- Alice samples \( K \leftarrow \text{Gen}(1^\lambda) \), and tells Eve \( \hat{a} \) (the msg length).

- Eve chooses distinct \( M_0, M_1 \) of equal length \( \lambda \).

- Alice chooses a random bit \( b \in \{0, 1\} \).

- Alice gives \( \text{Enc}(K, M_b) \) to Eve.

- Eve produces a guess \( \hat{b} \) for \( b \).

- Eve “wins” if \( \hat{b} = b \).

Eve’s advantage is \( \Pr[\hat{b} = b] - \frac{1}{2} \).

Advantage should go to zero as \( \lambda \) increases. (usually, we require advantage to be negl(\( \lambda \)).

Other security definitions:

- Known ciphertext attack
- Known CT/PT pairs
- Chosen PT
- Chosen CT

\{ assumes \( K \) is reused. \}
One-Time Pad (OTP)

Invented by Gilbert Vernam 1917: Paper-tape based (Patented).

Proposed a teleprinter cipher in which a previously prepared key, kept in paper tape (punched tape), is combined character-by-character w. plaintext msg to produce a ciphertext.

- Msg, Key, ciphertext have same length (× bits).
- Key K is called pad.

It is random & known only to Alice & Bob
(used by spies, key written on small pad)

\[
\text{Enc: } M = 101100 \\
\oplus K = 011010 \\
\underline{C = 110110} \quad \text{(mod } 2 \text{ each column)}
\]

\[
\text{Dec: Simply XOR K again} \\
(M_i \oplus K_i) \oplus K_i = M_i \oplus (K_i \oplus K_i) = M_i \oplus 0 = M_i
\]
Theorem [Shannon 49]: OTP is unconditionally secure
(i.e., secure against Eve w. unlimited computational power)

a.k.a. information theoretically secure.

[As opposed to computational security, which assumes Eve is computationally bounded.]

Proof: Recall (by def) Eve chooses \( M_0, M_1 \) of length \( \lambda \), as a bit \( b \in \{0,1\} \) is chosen at random.

Let \( \Pr[C] = \text{prob that the encryption that Eve receives is } C \).

\[ \Pr[C|b] = \text{prob that Eve receives ciphertext } C \text{ conditioned on } b \text{ being the chosen bit. (i.e. Eve receives enc.)} \]

Similarly: \( \Pr[b], \Pr[b|C] \)

We need to prove \( \forall b \in \{0,1\} \neq C \)

\[ \Pr[b|C] = \frac{1}{2} \quad \leftarrow \text{Perfect security!} \]

\[ \Pr[b|C] = \frac{\Pr[C|b] \cdot \Pr[b]}{\Pr[C]} = \frac{\Pr[C|b]}{\Pr[C]} \cdot \frac{1}{2} \]

Bayes' Rule.
Need to prove

\[ P_r[C] = P_r[C | b] \]

We will prove that they both equal \( \frac{1}{2^a} \):

\[ P_r[C | b] = P_r[K = C \oplus M_b] = \frac{1}{2^a} \]

\[ K \oplus M_b \]

\[ P_r[C] = P_r[C | b=0] \cdot P_r[b=0] + P_r[C | b=1] \cdot P_r[b=1] \]

\[ = \frac{1}{2^a} \cdot \frac{1}{2} + \frac{1}{2^a} \cdot \frac{1}{2} = \frac{1}{2^a} \]

This is perfect security!

Negatives:

- Generate large keys
- Share them securely
- Keep them secret
- Avoid reusing them!

\[ C_1 \oplus C_2 = (M_1 \oplus K) \oplus (M_2 \oplus K) = M_1 \oplus M_2 \]

* Google "Venona Project"
Note: OTP is malleable

Namely, $adv$ can (efficiently) change ciphertext bits, causing the decrypted msg to change.

$\Rightarrow$ OTP does not provide any authentication of msg or content, or protection against modification ("mauling").