Stream ciphers

Stream ciphers

Attacks on OTP and stream ciphers

Review

OTP: $E(k,m) = m \oplus k$, $D(k,c) = c \oplus k$

Making OTP practical using a PRG: $G: K \longrightarrow \{0,1\}^n$

Stream cipher: $E(k,m) = m \oplus G(k)$, $D(k,c) = c \oplus G(k)$

Security: PRG must be unpredictable (better def in two segments)

Attack 1: **two time** pad is insecure !!

Never use stream cipher key more than once !!

$C_1 \leftarrow m_1 \oplus PRG(k)$

$C_2 \leftarrow m_2 \oplus PRG(k)$

Eavesdropper does:

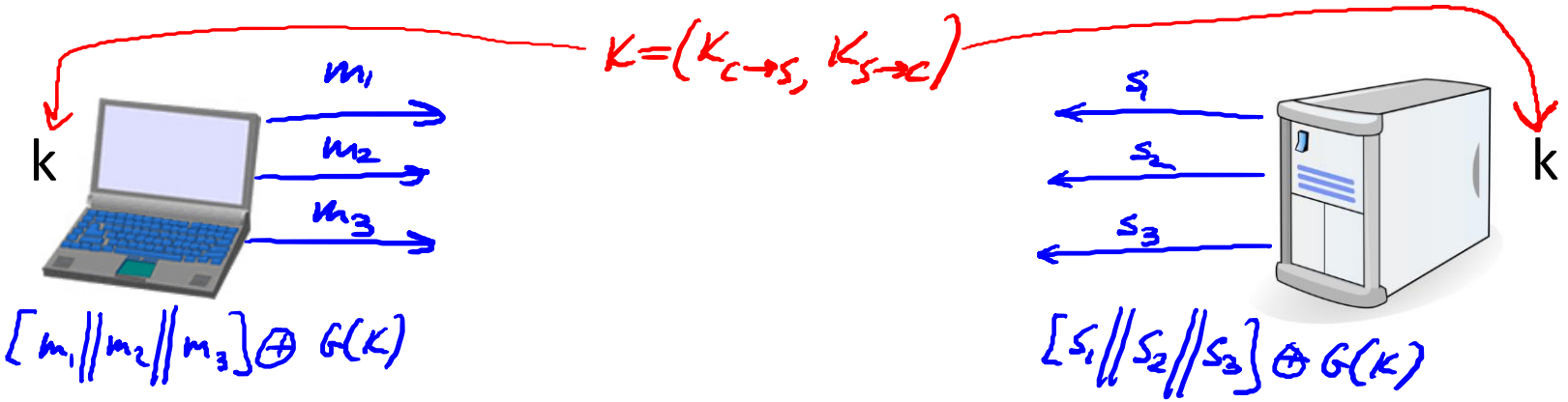
$C_1 \oplus C_2 \rightarrow m_1 \oplus m_2$

Enough redundancy in English and ASCII encoding that:

$m_1 \oplus m_2 \rightarrow m_1 , m_2$

Real world examples

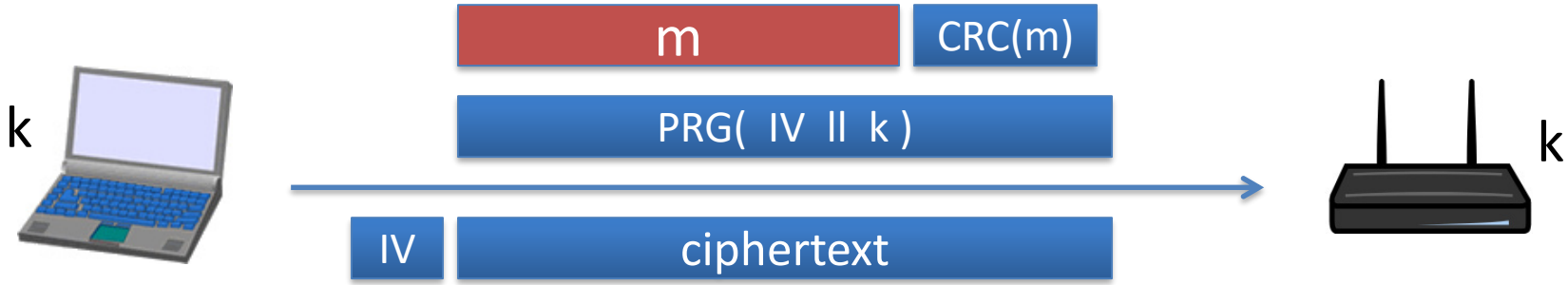
- MS-PPTP (windows NT):



Need different keys for C⟶S and S⟶C

Real world examples

802.11b WEP:

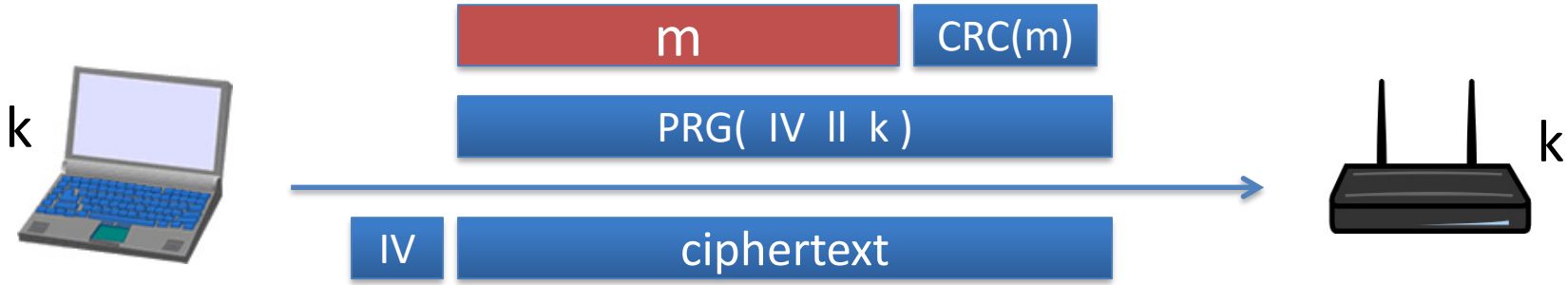


Length of IV: 24 bits

- Repeated IV after $2^{24} \approx$ 16M frames
- On some 802.11 cards: IV resets to 0 after power cycle

Avoid related keys

802.11b WEP:



key for frame #1: (1 ll k)

key for frame #2: (2 ll k)

⋮

24 bits

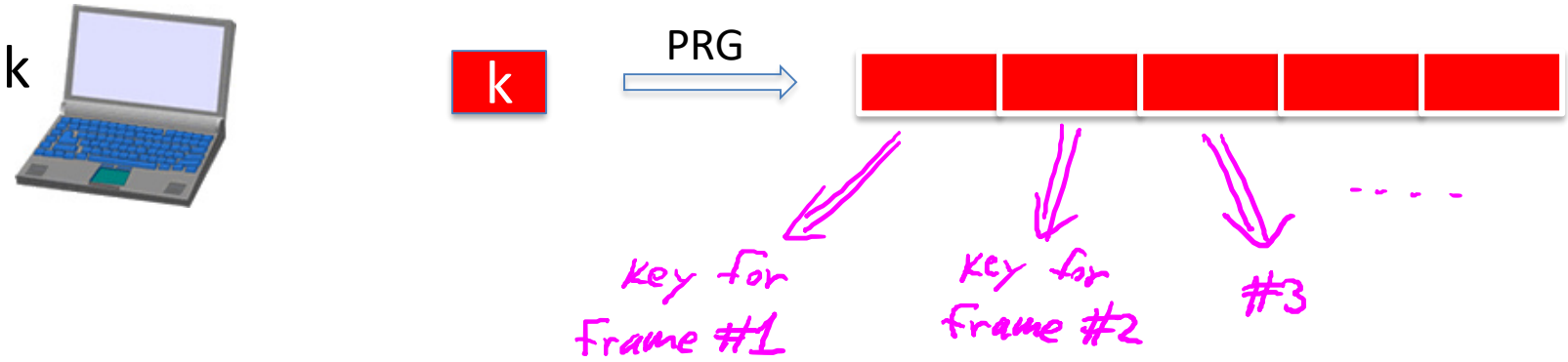
104 bits

For the RC4 PRG:

FMS2001 ⇒ can recover k after 10^6 frames

Recent attacks ≈ 40,000 frames

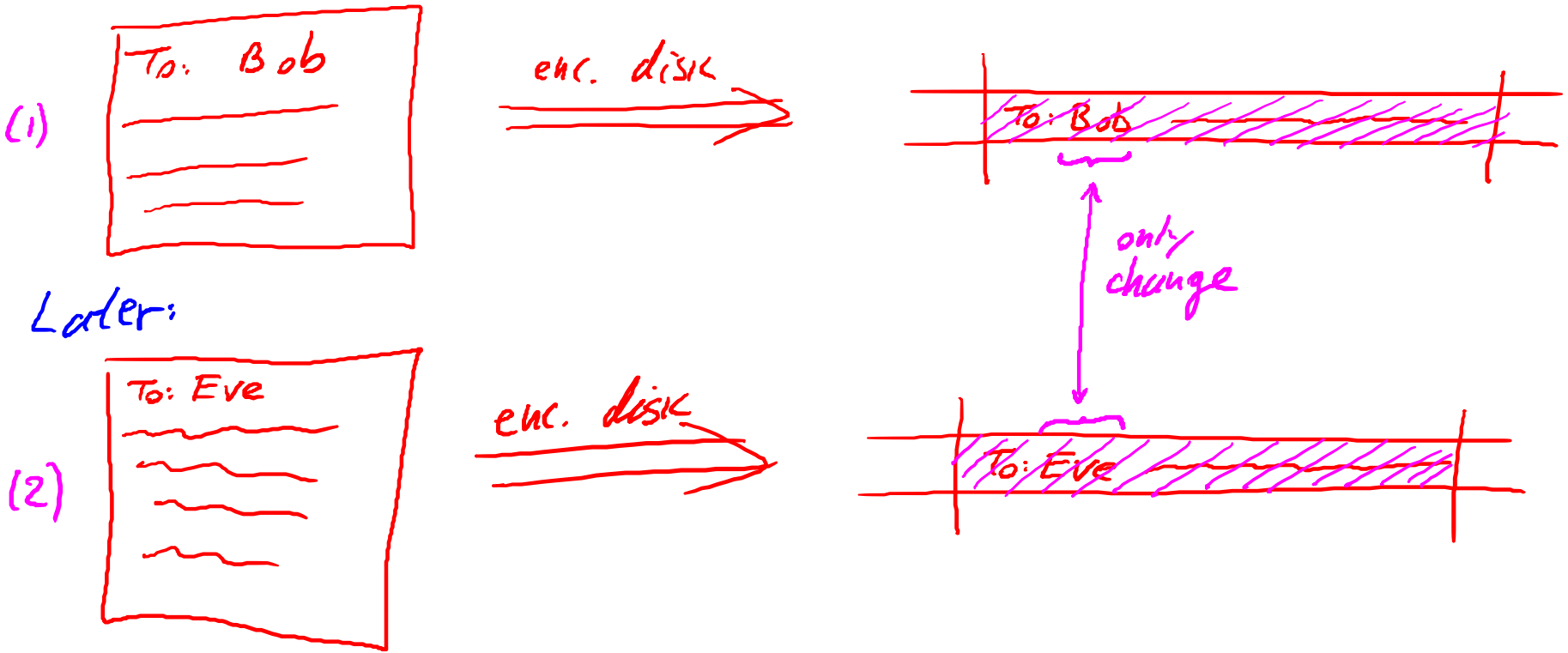
A better construction



⇒ now each frame has a pseudorandom key

better solution: use stronger encryption method (as in WPA2)

Yet another example: disk encryption

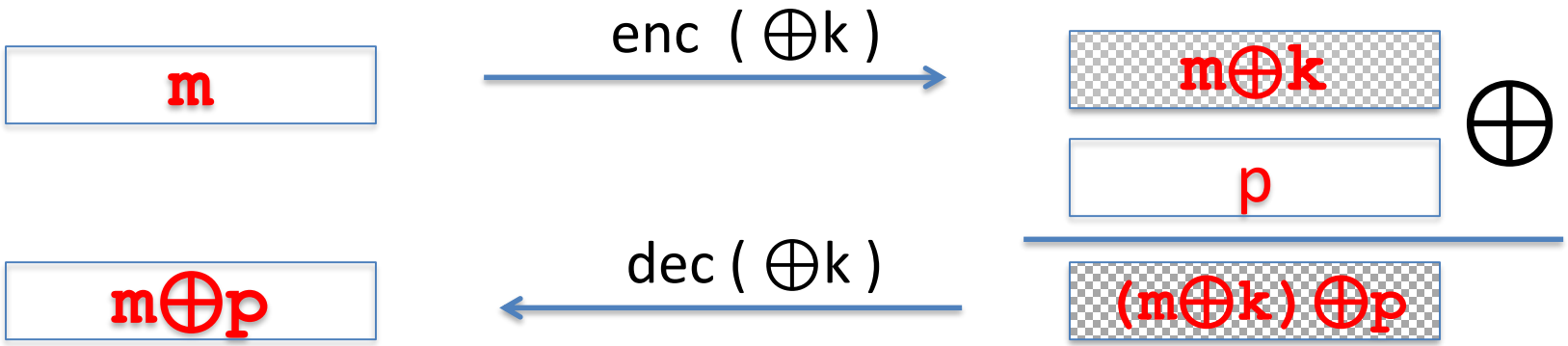


Two time pad: summary

Never use stream cipher key more than once !!

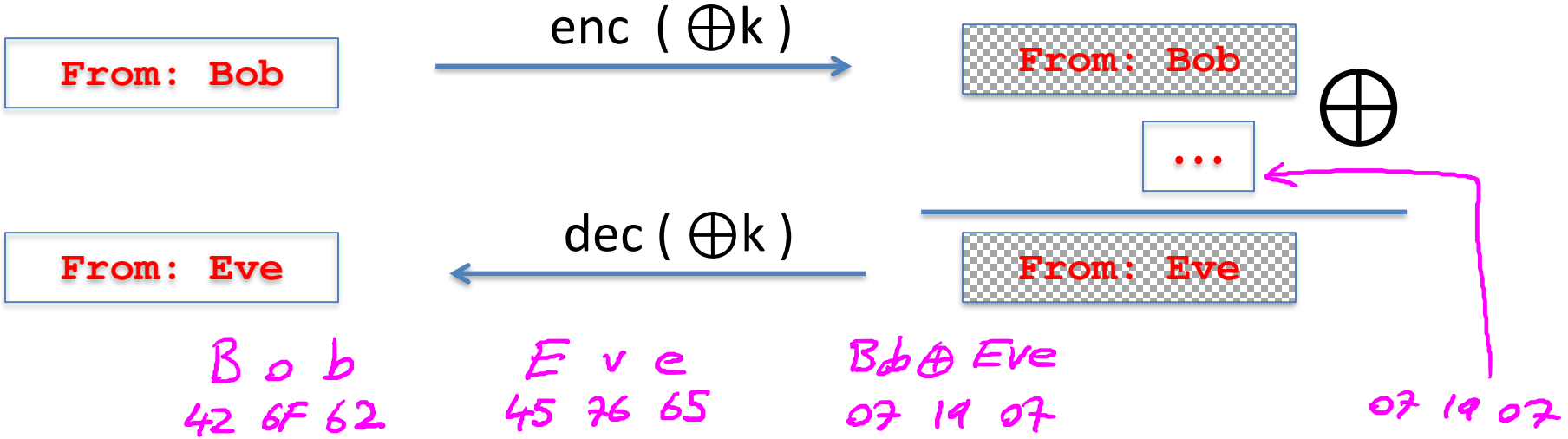
- Network traffic: negotiate new key for every session (e.g. TLS)
- Disk encryption: typically do not use a stream cipher

Attack 2: no integrity (OTP is malleable)



Modifications to ciphertext are undetected and have **predictable** impact on plaintext

Attack 2: no integrity (OTP is malleable)



Modifications to ciphertext are undetected and have predictable impact on plaintext

End of Segment

Stream ciphers

PRG Security Defs

Let $G: K \longrightarrow \{0,1\}^n$ be a PRG

Goal: define what it means that

$[k \xleftarrow{R} K,\ \text{output } G(k)]$

is "indistinguishable" from

$[r \xleftarrow{R} \{0,1\}^n,\ \text{output } r]$

Statistical Tests

Statistical test on $\{0,1\}^n$:

not random — random

an alg. A s.t. A(x) outputs "0" or "1"

Examples:

(1) $A(x)=1$ iff $\left| \#0(x) - \#1(x) \right| \leq 10 \cdot \sqrt{n}$

(2) $A(x)=1$ iff $\left| \#00(x) - \frac{n}{4} \right| \leq 10 \cdot \sqrt{n}$

Statistical Tests

More examples:

(3) $A(x)=1$ iff max-run-of-0$(x) < 10 \cdot \log_2(n)$

Advantage

Let $G: K \longrightarrow \{0,1\}^n$ be a PRG and A a stat. test on $\{0,1\}^n$

Define:

$$Adv_{PRG}[A,G] := \left| \Pr_{K \xleftarrow{R} K}[A(G(K))=1] - \Pr_{r \xleftarrow{R} \{0,1\}^n}[A(r)=1] \right| \in [0,1]$$

Adv close to 1 $\Longrightarrow$ A can dist. G from random

Adv close to 0 $\Longrightarrow$ A cannot

A silly example: $A(x) = 0 \Rightarrow Adv_{PRG}[A,G] = 0$

Suppose $G:K \longrightarrow \{0,1\}^n$ satisfies **msb(G(k)) = 1** for 2/3 of keys in K

Define stat. test A(x) as:

if [msb(x)=1] output "1" else output "0"

Then

$$Adv_{PRG}[A,G] = \left| Pr[A(G(k))=1] - Pr[A(r)=1] \right| =$$

$$\left| 2/3 - 1/2 \right| = 1/6$$

Secure PRGs: crypto definition

Def: We say that $G: K \longrightarrow \{0,1\}^n$ is a **secure PRG** if

$\forall$ "eff" stat. tests A:

$Adv_{PRG}[A, G]$ is "negligible"

Are there provably secure PRGs? Unknown (=> P != NP)

but we have heuristic candidates.

Easy fact: a secure PRG is unpredictable

We show: PRG predictable $\Rightarrow$ PRG is insecure

Suppose A is an efficient algorithm s.t.

$$\Pr_{k \xleftarrow{R} \{0,1\}^n}\left[A(G(k)|_{1,\ldots,i}) = G(k)|_{i+1} \right] > \frac{1}{2} + \varepsilon$$

for non-negligible ε (e.g. $\varepsilon = 1/1000$)

Easy fact: a secure PRG is unpredictable

Define statistical test B as:

$B(x) =$ if $A(x|_{1,\ldots,i}) = x_{i+1}$ output 1

else output 0

$r \xleftarrow{R} \{0,1\}^n$: $Pr[B(r)=1] = \frac{1}{2}$

$r \xleftarrow{R} \mathcal{K}$: $Pr[B(G(k))=1] > \frac{1}{2} + \varepsilon$

$\Rightarrow Adv_{PRG}[B,G] = \left| Pr[B(r)=1] - Pr[B(G(k))=1] \right| > \varepsilon$

Thm (Yao'82): an unpredictable PRG is secure

Let $G:K \longrightarrow \{0,1\}^n$ be PRG

"Thm": if $\forall i \in \{0, \ldots, n-1\}$ PRG G is unpredictable at pos. i
then G is a secure PRG.

If next-bit predictors cannot distinguish G from random
then no statistical test can !!

Let $G:K \longrightarrow \{0,1\}^n$ be a PRG such that

from the last n/2 bits of G(k)

it is easy to compute the first n/2 bits.

Is G predictable for some $i \in \{0, \ldots, n-1\}$?

- Yes
- No

Let $G: K \longrightarrow \{0,1\}^n$ be a PRG

Goal: define what it means that

$[k \xleftarrow{R} K,\ \text{output } G(k)]$

is "indistinguishable" from

$[r \xleftarrow{R} \{0,1\}^n,\ \text{output } r]$

Secure PRGs: crypto definition

Def: We say that $G: K \longrightarrow \{0,1\}^n$ is a **secure PRG** if

$\forall$ "eff" stat. tests A:

$Adv_{PRG}[A, G]$ is "negligible"

More Generally

Let P_1 and P_2 be two distributions over $\{0,1\}^n$

Def: We say that P_1 and P_2 are

computationally indistinguishable (denoted $P_1 \approx_p P_2$)

if $\forall$ "efficient" stat. tests A

$$\left| \Pr_{x \leftarrow P_1}[A(x)=1] - \Pr_{x \leftarrow P_2}[A(x)=1] \right| < \text{negligible}$$

Example: a PRG is secure if $\{ k \xleftarrow{R} K : G(k) \} \approx_p \text{uniform}(\{0,1\}^n)$

End of Segment