8) Steganography

Christian Cachin

Privacy and Data Security, 2021

(originally presented at CMS 2005)

Steganography ≠ Cryptography

- Cryptography hides the content of communication
  - Adversary knows that communication exists
  - Conceals data
- Steganography hides the existence of communication
  - Adversary should not discover existence of communication
  - Conceals also metadata

Information-hiding concepts

Embed information in a "cover" (carrier signal)

Steganography
- Hide presence of hidden information
- To avoid censorship and surveillance
- Presence not known; can easily be removed

Watermarking
- Hide information in a robust way
- To authenticate (multimedia) data, control ownership
- Presence is known; should be difficult to remove

Fingerprinting
- Like watermarking, but hides user-specific information for tracing and litigation

Prisoner's problem

Alice and Bob want to coordinate their escape

Communication through passive observer (warden, censor ...)

- innocent communication is allowed
- talking about escape plans is forbidden

Formulated by Simmons (1983)

Steganography = hidden communication

How to be convinced that it is "really" hidden?

Many attacks

- Correlation, histogram, transforms ...

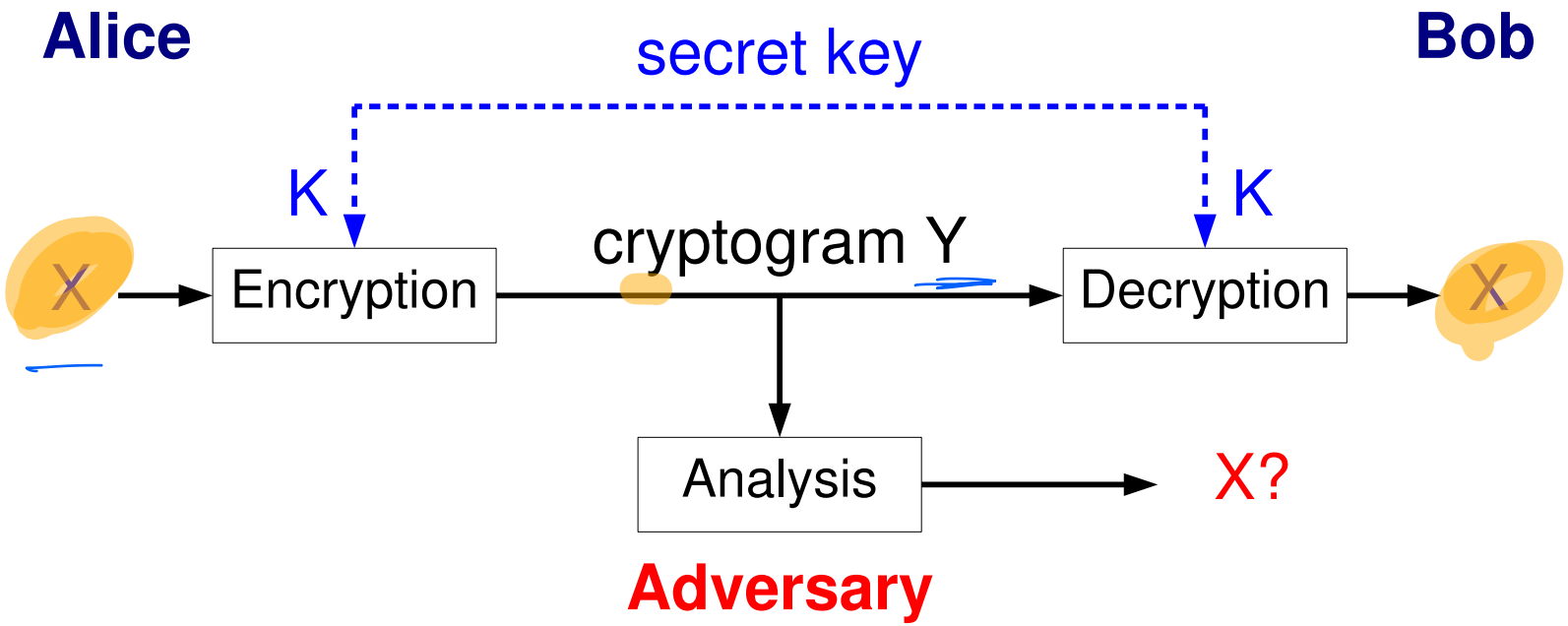
Analogy to cryptology

- Ages of broken cryptosystems until ~1980
- Theory of cryptology with provably secure cryptosystems since ~1990

→ Formal model for steganography

Models for cryptosystems

Model of a cryptosystem



Shannon (1949)

Adversary is *passive*

Security of cryptosystems

Perfect security (Shannon 1949)

$$I(X;Y) = H(X) - H(X|Y) = 0$$

Information-theoretic
Unbounded adversary obtains no information
Implies also that $H(K) \geq H(X)$ ☹
Ex. one-time pad

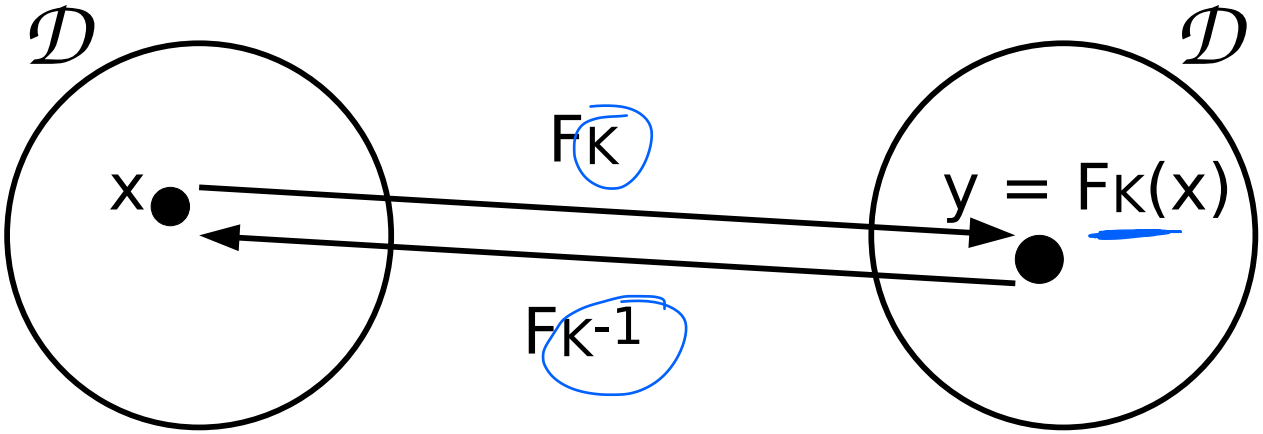
But ... practical cryptography uses block- and stream-ciphers with short keys.

Practical cryptosystems

Family of one-way permutations

$\mathcal{F} = \{F_K\}, F_K : \mathcal{D} \rightarrow \mathcal{D}$

indexed by key K

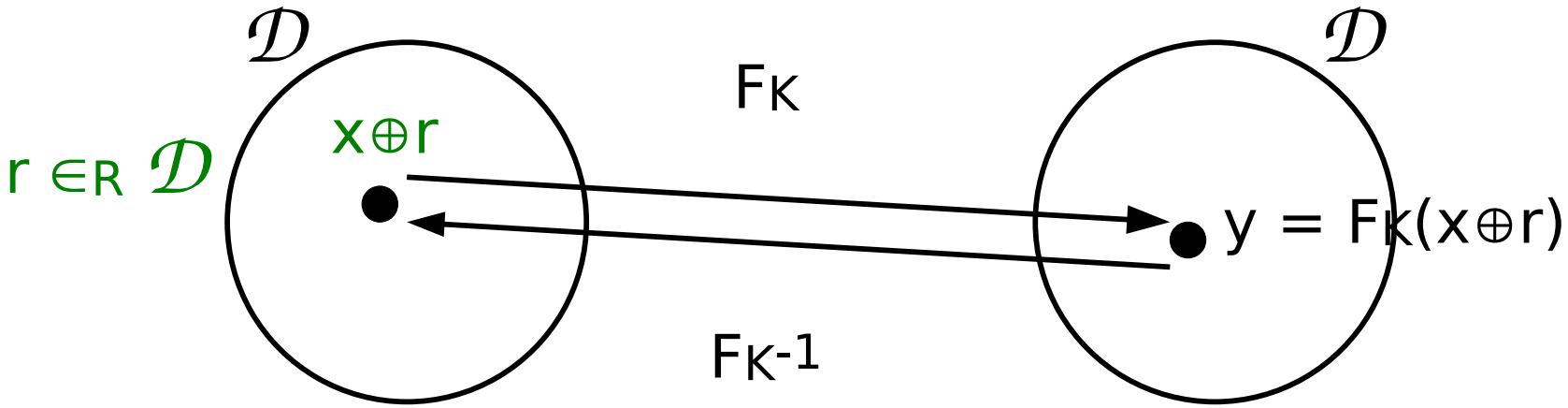


Encryption and decryption are deterministic

leaks information

Probabilistic encryption

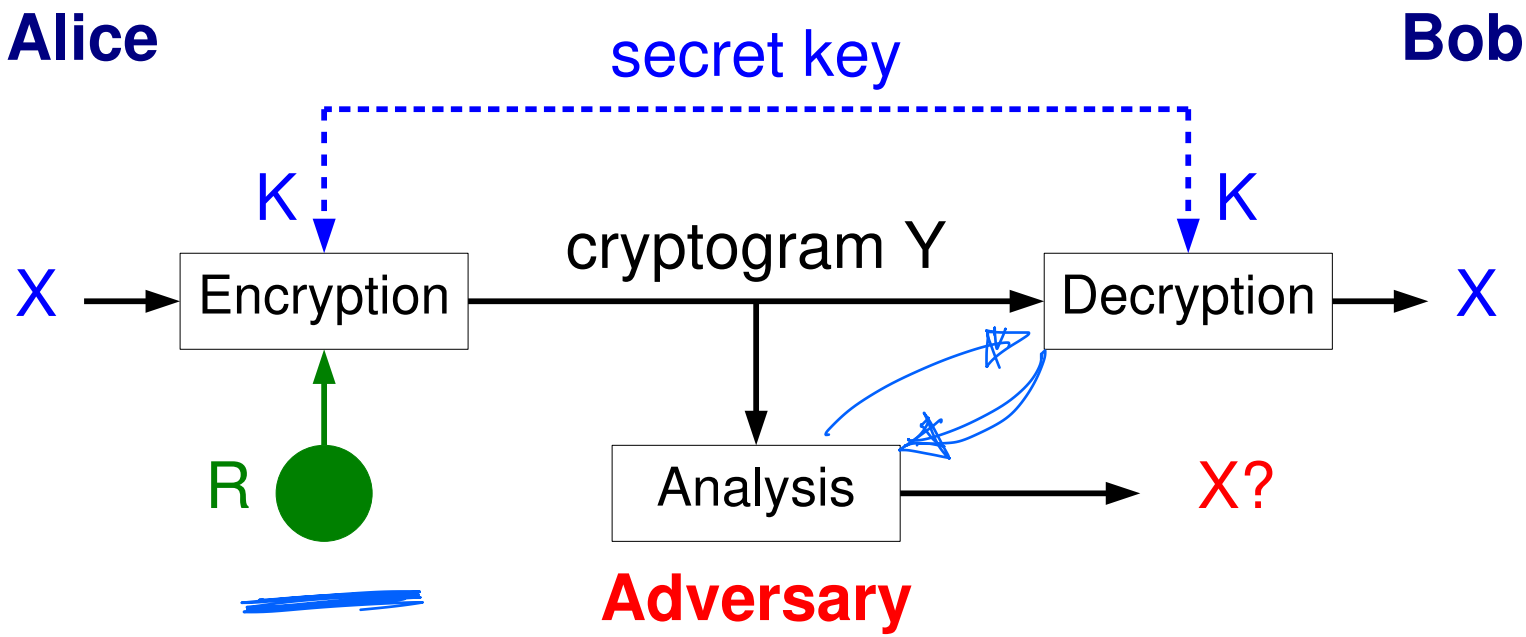
Randomize input to encryption function



Ciphertext is (r, y)

- does not repeat for same plaintext x
- better protection

Probabilistic cryptosystems



Private random source R

Computational security for cryptosystems

Formal security model (Goldwasser-Micali, 1985)

- Chosen-plaintext attacks (passive adversary)
- Semantic security ⇔ indistinguishability of ciphertexts

Defined by experiment with adversary A, a probabilistic polynomial-time (PPT) algorithm:

$K \leftarrow KG(1^k)$

$(x_0,x_1) \leftarrow A_1(K)$

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

$c^* \leftarrow E(K,x_b)$

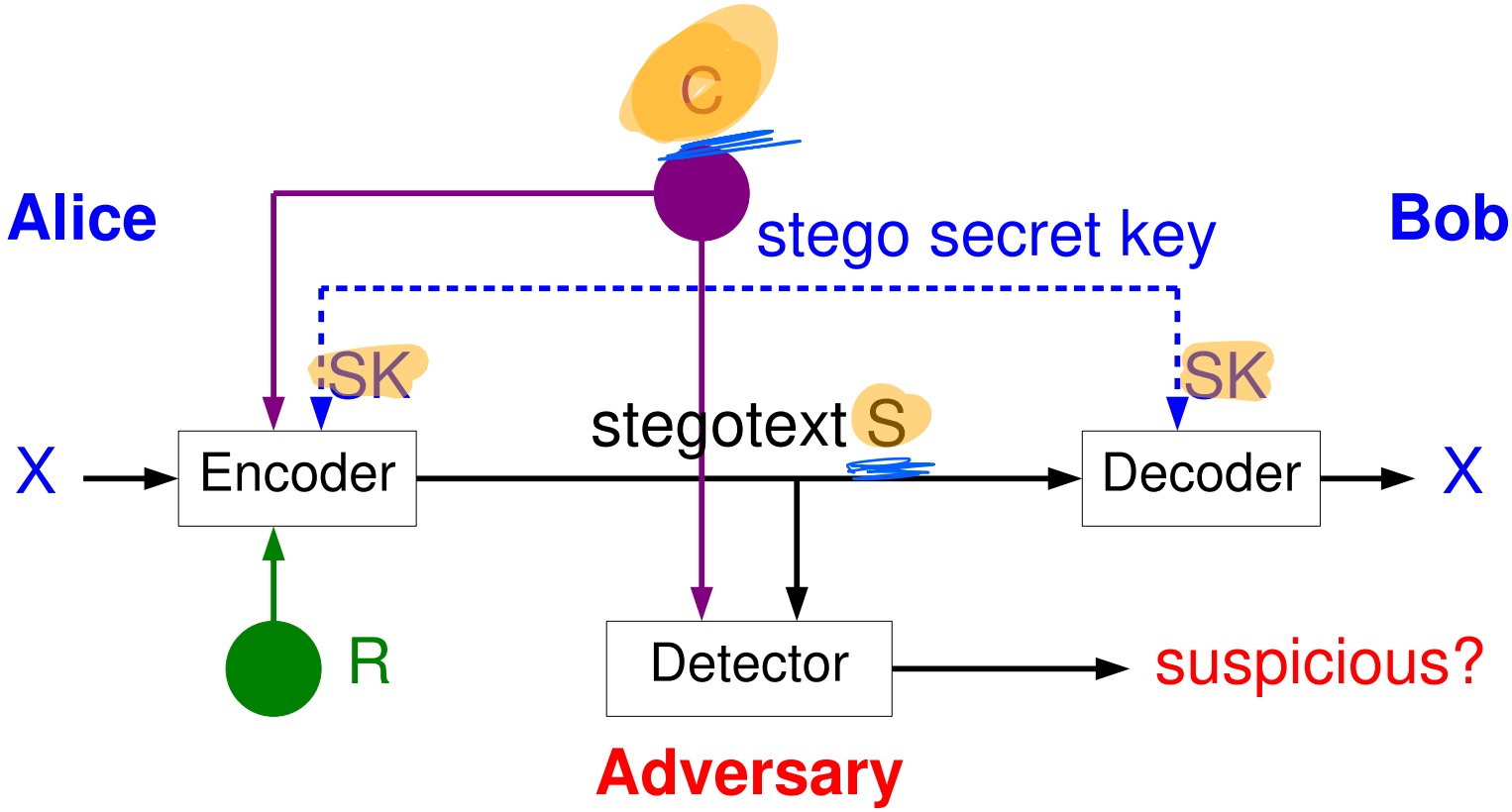
$b^* \leftarrow A_2(x_0,x_1,c^*)$

Then

$\forall$ PPT $A=(A_1,A_2)$: $Pr[\ b = b^*\ ] \leq ½ + negl.$

Models for stegosystems

Model of a stegosystem



Covertext C and stegotext S over domain Y

Adversary is *passive*

How to define security for stegosystems?

Attempt 1

Based on information theory and mutual entropy:

$$I(X;S|C) = 0$$

Fails to differentiate stegosystem from cryptosystem.

Attempt 2

Based on distortion measure $d(C,S) = \|P_C - P_S\|_2$:

$$d(c,s) \rightarrow 0$$

Fails for some examples, e.g., when C is a random n-bit string, but S is a random n-bit string with even parity.

Information-theoretic security

Steganography as hypothesis testing (C. 1998)

Adversary distinguishes stegotext from covertext

Quantified using relative entropy

Perfect security

$$D(P_C \| P_S) = 0$$

Statistical security

$$D(P_C \| P_S) \leq \varepsilon$$

Discrimination or relative entropy

$$D(P_C \| P_S) = \sum_{y \in Y} P_C(y) \log_2(P_C(y)/P_S(y)) \geq 0$$

$\mathcal{A}^C \qquad \mathcal{A}^S$

Bounds on detection

Adversary distinguishes stegotext S (H_1) from covertext C (H_0)

Deciding S with signal C is type-I error, prob. α

Deciding C with signal S is type-II error, prob. β

Statistical security $D(P_C \| P_S) \leq \varepsilon$,

$d(\alpha,\beta) \leq \varepsilon$

binary discrimination $d(\bullet,\bullet)$

For $\alpha=0$, this implies

$\beta \geq 2^{-\varepsilon}$

Cover & Thomas textbook on information theory

Modeling the covertext

Probabilistic source or channel

- **C** = C_0, C_1, C_2 ...
- Distribution known

C is a stochastic process

- Random variable (i.i.d. → ergodic → ...)

Unrealistic in practice

- Real-world communication does not come with specification of distribution

Universal stegosystems

What if distribution of **C** is not known?

→ Universal stegosystems

→ No knowledge of cover distribution needed

C is an algorithm, given as oracle

Can be queried on arbitrary history

$\mathbf{C}(h,n) = C_{|h|+1}, C_{|h|+2}, \ldots, C_{|h|+n}$

Synthetic cover signal **C** from machine learning

Example stegosystem

Known distribution P_C over domain Y

Message $x \in \{0,1\}$

Let

$Y_0 = \min_{Y' \subseteq Y} |\Pr[C \in Y'] - \Pr[C \notin Y']|$ (prob. =: ε)

$Y_1 = Y \setminus Y_0$

Key $sk \in_R \{0,1\}$

RV C_0 is C restricted to Y_0

RV C_1 is C restricted to Y_1

Stego-encoder: $SE(sk,x) = C_{x \oplus sk}$

Stego-decoder: $SD(sk,y) = 0$ iff $y \in Y_{sk}$

Thm.: This is an $\varepsilon^2/\ln 2$-statistically secure stegosystem (in terms of relative entropy).

Computationally secure stegosystems

Analogous to computational security model for cryptosystems

- Universal stegosystem
  - covertext given as oracle **C**
- Chosen-plaintext attacks (CPA)
  - passive adversary
- Indistinguishability of covertext and stegotext

Secret-key stegosystem: SKG, SE, SD

- SKG(1^k) → sk
- SE(sk,x) → c
- SD(sk,c) → x or ⊥

Computational stegosystems (2)

Robustness

$SD(sk, SE(sk,x)) = x$

SS-CPA security defined by experiment with SA

$K \leftarrow SKG(1^k)$

$(x^*,s) \leftarrow SA_1(K)$

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

if b=0 then $c^* \leftarrow SE(K,x^*)$ else $c^* \leftarrow_R \mathbf{C}$ fi

$b^* \leftarrow SA_2(x^*,c^*,s)$

Then

$\forall$ PPT $SA=(SA_1,SA_2) : Pr[\ b = b^*\ ] \leq ½ + negl.$

Stegotext and covertext are indistinguishable

Construction – Assumptions

Given

Sym. cryptosystem (KG,E,D) with pseudo-random ciphertexts & universal hash function G

SKG(1^k)

Run KG(1^k) → sk and pick $g \in_R G$ → (sk,g)

Construction – Rejection sampler

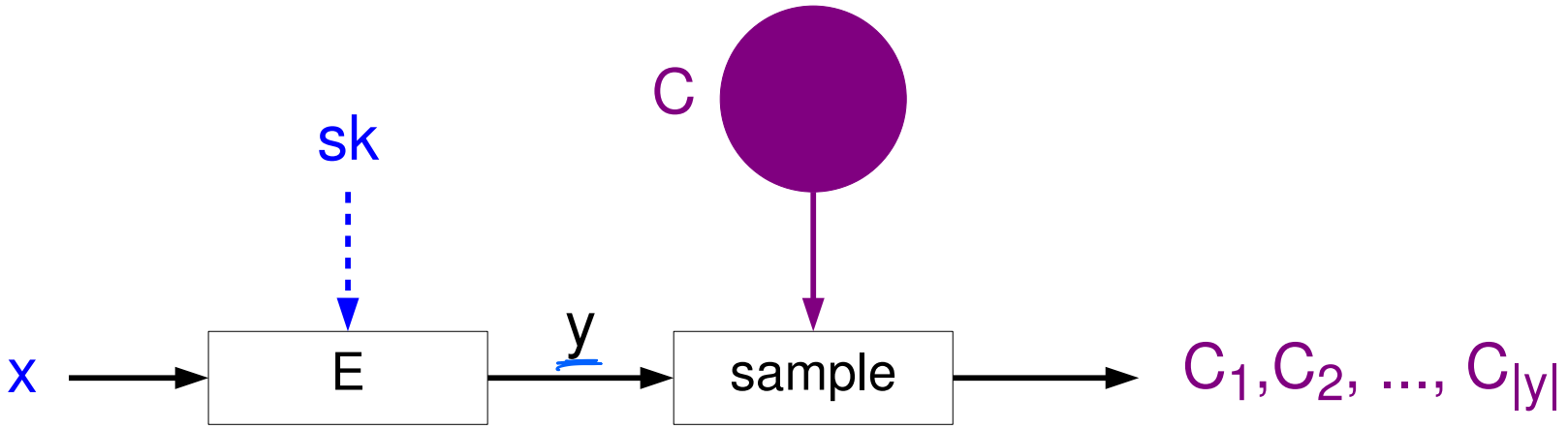
SE(sk,m)
  z ← E(sk,x)
  for i = 1 ... |z| **do**
    repeat
      sample $C_i \leftarrow_R$ **C**
    until $g(C_i) = z_i$
  output $C_1 \dots C_{|z|}$

SD(sk, $C_1 \dots C_{|z|}$)
  for i = 1 ... |z| **do**
    $z_i \leftarrow g(C_i)$
  x ← D(sk,z)

$g : C \rightarrow \{0,1\}$

hash function

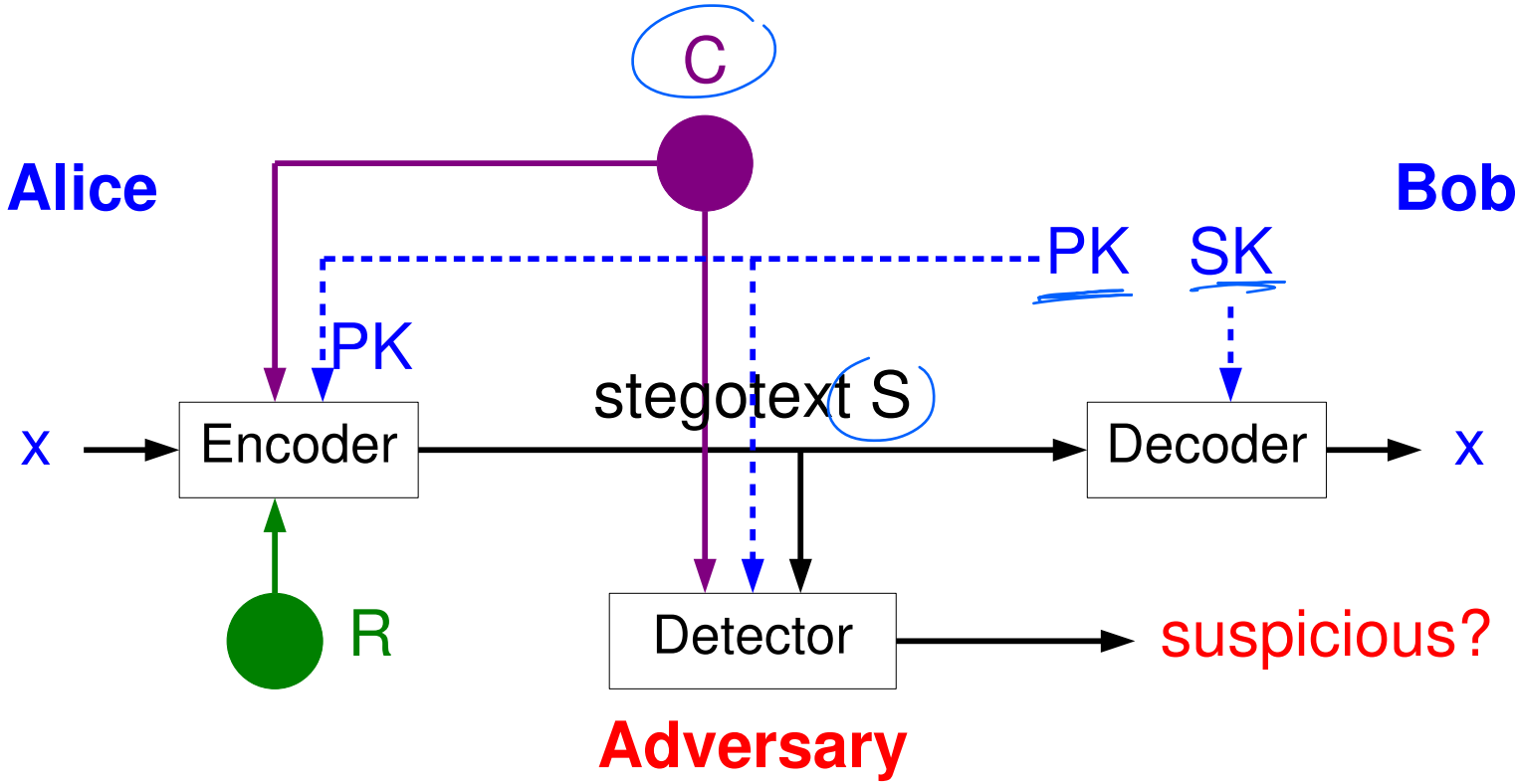
Rejection sampler



If **C** has large enough min-entropy, then

$\| \langle g, C_1, C_2 \ldots C_{|y|} \rangle - \langle g, \mathbf{C}^{|y|} \rangle \| \leq$ negl.

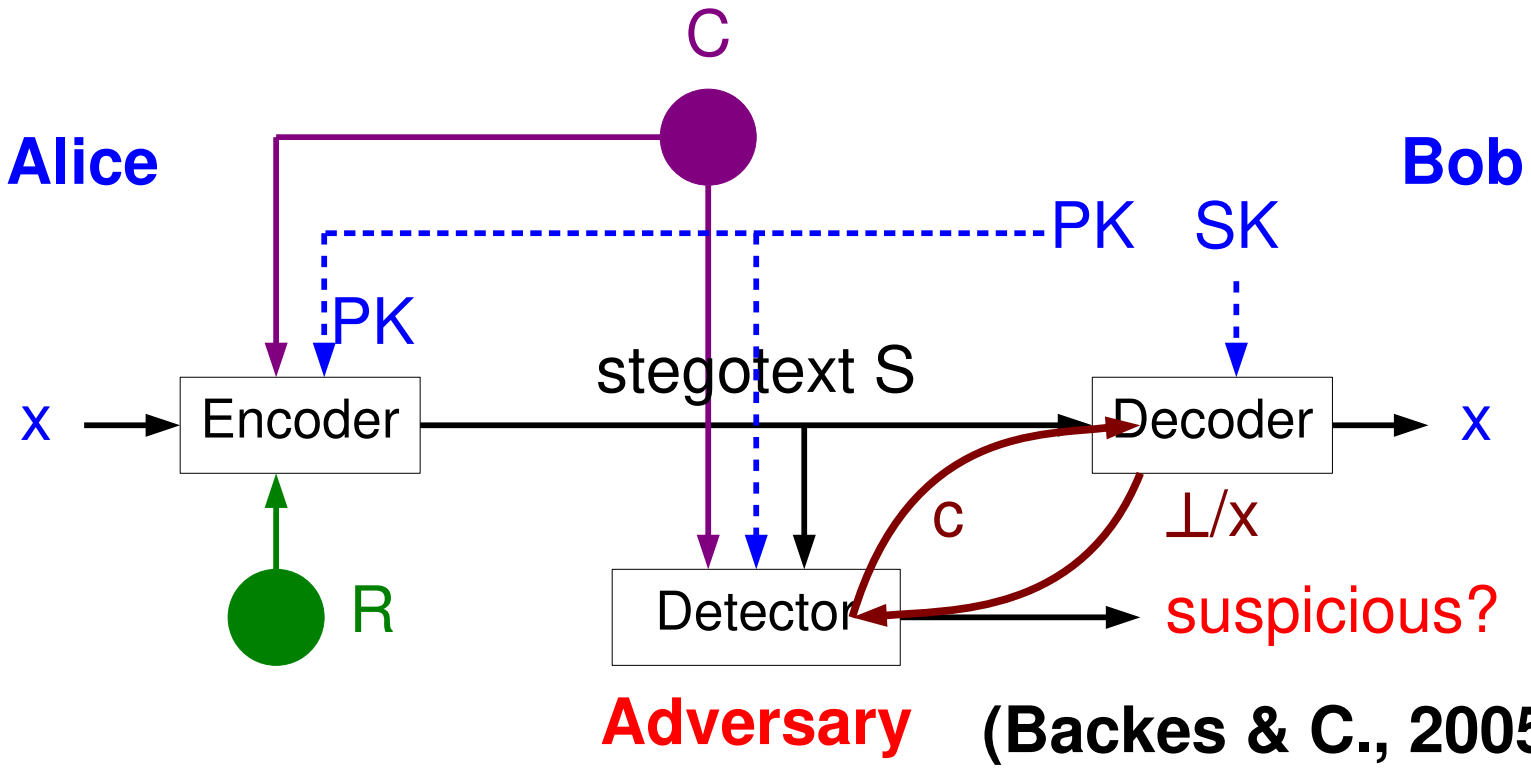
Public-key stegosystems



(v. Ahn & Hopper, 2004)

Bob has public-key/secret-key pair (PK,SK)
Adversary is passive (CPA) or active (CCA)

Modeling active attacks



Adversary may ask Bob if he considers c to be stegotext
Bob answers 0 (cover) or 1 (stego)

Analogous to adaptive chosen-ciphertext attacks for public-key cryptosystems

Practical steganography

Many tools for image and audio steganography

- Ca. 1990-2005
- Starting from making imperceptible modifications to the least significant bit (LSB, in pixel or audio data)

Not relevant in practice today

- Steganography is often very easy to detect with forensic image analysis tools

Recently – New ideas from machine learning

Stegosystems using ML

Meteor (2021)

- https://meteorfrom.space
- https://eprint.iacr.org/2021/686

Sampler uses ML to generate realistic covers

- Natural-language text
- Uses OpenAI's GPT-2 (Generative Pre-trained Transformer) that produces human-like texts

Meteor

- Examples 1-3
- Long explanation: https://meteorfrom.space

Summary

Definition of stegosystems

Security for (secret-key) stegosystems
- Perfect → statistical → computational
- Always relative to covertext distribution

Public-key stegosystems
- Computational security

Realistic stegosystems using machine learning
- Automatically generate covertext
- Permitted covertext distribution is critical

Executive summary

Stegosystems are cryptosystems with prescribed distribution of ciphertext