Introduction to Cryptology

2.3 - Computational Secrecy

Federico Pintore

Mathematical Institute, University of Oxford (UK)

Michaelmas term 2020

Computational Security

Definition (Concrete version)

A scheme is secure if any adversary $\mathcal{A}$

- running for time at most t
- succeeds in breaking the scheme with probability at most ϵ.

[1] Both the running time and the success probability are expressed as functions of the security parameter n.

Computational Security

Definition (Concrete version)

A scheme is secure if any adversary $\mathcal{A}$

- running for time at most t
- succeeds in breaking the scheme with probability at most ϵ.

Definition (Asymptotic version)

A scheme is asymptotically secure if any probabilistic polynomial-time (in n) adversary $\mathcal{A}$ succeeds in breaking the scheme with at most negligible probability (in n)[1].

[1] Both the running time and the success probability are expressed as functions of the security parameter n.

Notation

Polynomial-time algorithm in n: its running time $f(n)$ is in $\mathcal{O}(n^\ell)$ for some $\ell \in \mathbb{N}$, i.e. $\exists N, \lambda \in \mathbb{N}$ s.t. $f(n) \leq \lambda n^\ell \quad \forall n \geq N$.

Notation

Polynomial-time algorithm in n: its running time $f(n)$ is in $\mathcal{O}(n^\ell)$ for some $\ell \in \mathbb{N}$, i.e. $\exists N, \lambda \in \mathbb{N}$ s.t. $f(n) \leq \lambda n^\ell \quad \forall n \geq N$.

Negligible function $g(n)$: for each $\ell \in \mathbb{N}$, there exists $N \in \mathbb{N}$ s.t.

$$g(n) \leq \frac{1}{n^\ell} \quad \forall n \geq N.$$

Notation

Polynomial-time algorithm in n: its running time $f(n)$ is in $\mathcal{O}(n^\ell)$ for some $\ell \in \mathbb{N}$, i.e. $\exists N, \lambda \in \mathbb{N}$ s.t. $f(n) \leq \lambda n^\ell \quad \forall n \geq N$.

Negligible function $g(n)$: for each $\ell \in \mathbb{N}$, there exists $N \in \mathbb{N}$ s.t.

$$g(n) \leq \frac{1}{n^\ell} \quad \forall n \geq N.$$

Probabilistic Algorithm: it has access to a random source.

Perfect Indistinguishability

Perfect Indistinguishability Experiment $\mathrm{PrivK}^{\mathrm{perfect-ind}}_{\mathcal{A},E}$

Challenger Ch		Adversary $\mathcal{A}$
	$\xleftarrow{m_0, m_1, \lvert m_0\rvert = \lvert m_1\rvert}$	
$b \leftarrow_{\$} \{0,1\}$		
	$\xrightarrow{c = \mathrm{Enc}(k, m_b)}$	
		Outputs their guess b'

Definition

An encryption scheme E is perfectly indistinguishable if, for every adversary $\mathcal{A}$, the following holds:

$$\Pr(\mathrm{PrivK}^{\mathrm{perfect-ind}}_{\mathcal{A},E} = 1) = 1/2,$$

where $\mathrm{PrivK}^{\mathrm{perfect-ind}}_{\mathcal{A},E} = 1$ if $b' = b$, and 0 otherwise.

Computational Indistinguishability

Adversarial Indistinguishability Experiment $\mathrm{PrivK}^{\mathrm{eav}}_{\mathcal{A},E}$

Challenger Ch

Adversary $\mathcal{A}$

$m_0, m_1, |m_0| = |m_1|$ ←

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

$c = \mathrm{Enc}(k, m_b)$ →

Outputs their guess b'

Computational Indistinguishability

Adversarial Indistinguishability Experiment $\mathrm{PrivK}^{\mathrm{eav}}_{\mathcal{A},E}$

Challenger Ch | Adversary $\mathcal{A}$

$\xleftarrow{m_0, m_1, |m_0|=|m_1|}$

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

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

Outputs their guess b'

Definition

An encryption scheme E is computationally indistinguishable if, for every PPT adversary $\mathcal{A}$, there exists a negligible function $\mathrm{negl}(n)$ s.t.

$$\Pr(\mathrm{PrivK}^{\mathrm{eav}}_{\mathcal{A},E} = 1) \leq \frac{1}{2} + \mathrm{negl}(n),$$

where $\mathrm{PrivK}^{\mathrm{eav}}_{\mathcal{A},E} = 1$ if $b' = b$, and 0 otherwise.

Computational Indistinguishability

- Does a computationally indistinguishable symmetric-key encryption scheme exist?

Computational Indistinguishability

- Does a computationally indistinguishable symmetric-key encryption scheme exist?

- Does a computationally indistinguishable symmetric-key encryption scheme with $|\mathcal{K}| \leq |\mathcal{M}|$ exist?

Computational Indistinguishability

- Does a computationally indistinguishable symmetric-key encryption scheme exist?

- Does a computationally indistinguishable symmetric-key encryption scheme with $|\mathcal{K}| \leq |\mathcal{M}|$ exist?

We could use pseudo-random generators to transform a random *short* key into a *random − looking* longer key...

Further Reading I

Nadhem J AlFardan, Daniel J Bernstein, Kenneth G Paterson, Bertram Poettering, and Jacob CN Schuldt.
On the security of RC4 in TLS.
In 22nd USENIX Security Symposium (USENIX Security 13), pages 305–320, 2013.

Boaz Barak and Shai Halevi.
A model and architecture for pseudo-random generation with applications to/dev/random.
In Proceedings of the 12th ACM conference on Computer and communications security, pages 203–212. ACM, 2005.

Daniel J Bernstein.
The Salsa20 Family of Stream Ciphers.
In New stream cipher designs, pages 84–97. Springer, 2008.

Further Reading II


Lenore Blum, Manuel Blum, and Mike Shub.
A simple unpredictable pseudo-random number generator.
SIAM Journal on computing, 15(2):364–383, 1986.


Christian Cachin.
Entropy measures and unconditional security in cryptography.
PhD thesis, ETH Zurich, 1997.


Scott Fluhrer, Itsik Mantin, and Adi Shamir.
Weaknesses in the key scheduling algorithm of RC4.
In Selected areas in cryptography, pages 1–24. Springer, 2001.

Further Reading III

Christina Garman, Kenneth G Paterson, and Thyla Van der Merwe.
Attacks only get better: Password recovery attacks against RC4 in TLS.
In 24th USENIX Security Symposium (USENIX Security 15), pages 113–128, 2015.

Itsik Mantin and Adi Shamir.
A practical attack on broadcast RC4.
In Fast Software Encryption, pages 152–164. Springer, 2002.