Concrete Properties of Hash Functions
Symmetric-Key Wrap-Up

CS/ECE 407

Today's objectives

Discuss concrete properties of hash functions

Define collision resistance and other concrete properties

Prove RO achieves these properties, discuss relationship between them

Review symmetric-key cryptography, and look forward to public-key cryptography

Random Oracle Model

Alice **Bob**

There's a public function black-box function H, and on any input H returns a random output

This is a bit like changing the parties' model of computation

before, parties were simply poly-bounded algorithms — we were in the **standard model**

In ROM, parties are poly-bounded computations *with access to global service **H***

Alice

Bob

Today

Can we define notions of hash functions that make sense *without* an idealized model?

Appears in *Fast Software Encryption(FSE 2004)*, Lecture Notes in Computer Science, Vol. 3017, Springer-Verlag.
This is the full version.

Cryptographic Hash-Function Basics: Definitions, Implications, and Separations for Preimage Resistance, Second-Preimage Resistance, and Collision Resistance

P. Rogaway [*] T. Shrimpton [†]

July 16, 2009

Abstract

We consider basic notions of security for cryptographic hash functions: collision resistance, preimage resistance, and second-preimage resistance. We give seven different definitions that correspond to these three underlying ideas, and then we work out all of the implications and separations among these seven definitions within the concrete-security, provable-security framework. Because our results are concrete, we can show two types of implications, *conventional* and *provisional*, where the strength of the latter depends on the amount of compression achieved by the hash function. We also distinguish two types of separations, *conditional* and *unconditional*. When constructing counterexamples for our separations, we are careful to preserve specified hash-function domains and ranges; this rules out some pathological counterexamples and makes the separations more meaningful in practice. Four of our definitions are standard while three appear to be new; some of our relations and separations have appeared, others have not. Here we give a modern treatment that acts to catalog, in one place and with carefully-considered nomenclature, the most basic security notions for cryptographic hash functions.

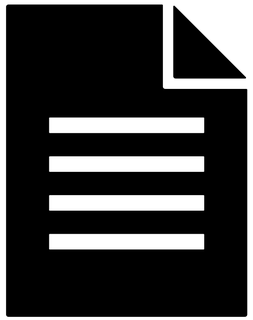
Key words: collision resistance, cryptographic hash functions, preimage resistance, provable security, second-preimage resistance.

[*] Dept. of Computer Science, University of California, Davis, California 95616, USA; and Dept. of Computer Science, Faculty of Science, Chiang Mai University, 50200 Thailand. E-mail: rogaway@cs.ucdavis.edu WWW: www.cs.ucdavis.edu/~rogaway/

[†] Dept. of Computer Science, Portland State University, Portland, Oregon 97201, USA. E-mail: teshrim@cs.pdx.edu WWW: www.cs.pdx.edu/~teshrim/

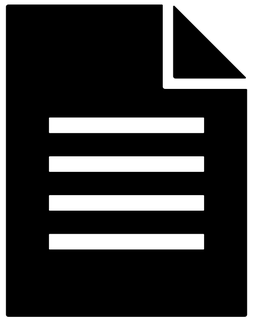
Alice

Bob



A.txt

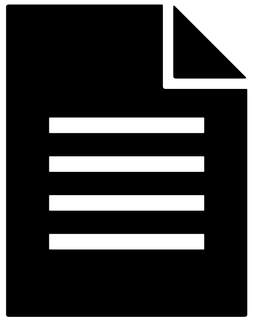
Are these files the same?



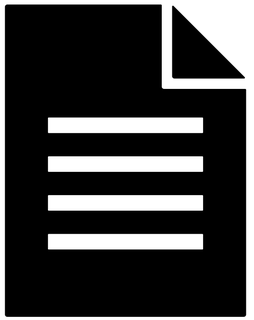
B.txt

Alice

Bob



Are these files the same?



A.txt

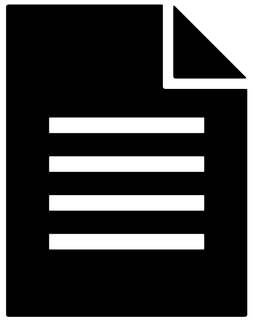
B.txt

Do we really need random oracle for this use-case?

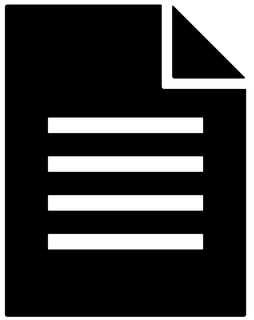
We just need that collisions are uncommon!

Alice

Bob



Are these files the same?



A.txt

B.txt

Do we really need random oracle for this use-case?

Consider a family of hash functions

$$H : \{0,1\}^{\lambda} \times \{0,1\}^{*} \rightarrow \{0,1\}^{\lambda}$$

A hash family H is **collision resistant** if no poly-time adversary can produce a hash collision

Namely, for any poly-time adversary A, the following probability is negligible

$$\Pr\left[H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^{\lambda} \\ (x_0, x_1) \leftarrow A(s) \end{array} \right] < \mathrm{negl}(\lambda)$$

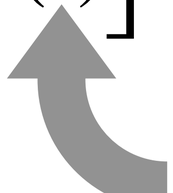
Consider a family of hash functions

$$H : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^\lambda$$

A hash family H is **collision resistant** if no poly-time adversary can produce a hash collision

Namely, for any poly-time adversary A, the following probability is negligible

$$\Pr\left[H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ (x_0, x_1) \leftarrow A(s) \end{array} \right] < \mathrm{negl}(\lambda)$$



Adversary gets the seed!

Random Oracles are Collision Resistant

$$H(s, x) = RO(s||x)$$

$$\Pr\left[H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^{\lambda} \\ (x_0, x_1) \leftarrow A^{RO}(s) \end{array}\right] < \text{negl}(\lambda)$$

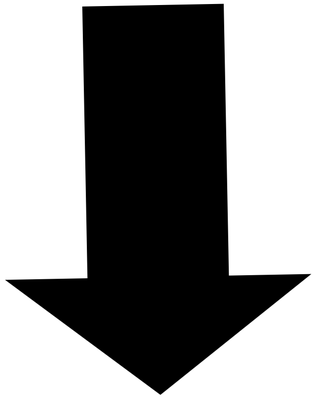
Collision Resistance:

$$\Pr\left[H(s,x_0) = H(s,x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ (x_0,x_1) \leftarrow A(s) \end{array}\right] < \mathrm{negl}(\lambda)$$

Useful, e.g., for MACs:
Adversary cannot forge a tag

Collision Resistance:

$$\Pr\left[H(s,x_0)=H(s,x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ (x_0,x_1) \leftarrow A(s) \end{array}\right] < \mathrm{negl}(\lambda)$$



Second-preimage resistance:

$$\Pr\left[H(s,x_0)=H(s,x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ x_0 \leftarrow \{0,1\}^\lambda \\ x_1 \leftarrow A(s,x_0) \end{array}\right] < \mathrm{negl}(\lambda)$$

Useful, e.g., for MACs on random messages:
Adversary cannot forge a tag

Second-preimage resistance:

$$\Pr\left[\, H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ x_0 \leftarrow \{0,1\}^\lambda \\ x_1 \leftarrow A(s, x_0) \end{array} \right] < \mathrm{negl}(\lambda)$$

Second-preimage resistance:

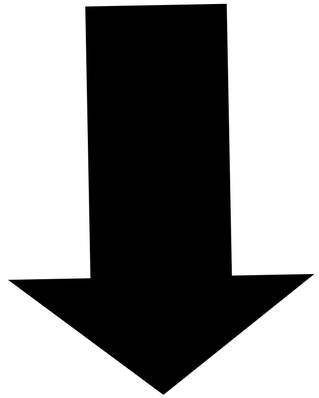
$$\Pr\left[\, H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ x_0 \leftarrow \{0,1\}^\lambda \\ x_1 \leftarrow A(s, x_0) \end{array} \right] < \mathrm{negl}(\lambda)$$

Preimage resistance:

$$\Pr\left[\, H(s, x) = y \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ y \leftarrow \{0,1\}^\lambda \\ x \leftarrow A(s, y) \end{array} \right] < \mathrm{negl}(\lambda)$$

Second-preimage resistance:

$$\Pr\left[\, H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ x_0 \leftarrow \{0,1\}^\lambda \\ x_1 \leftarrow A(s, x_0) \end{array} \right] < \mathrm{negl}(\lambda)$$



Preimage resistance:

$$\Pr\left[\, H(s, x) = y \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ y \leftarrow \{0,1\}^\lambda \\ x \leftarrow A(s, y) \end{array} \right] < \mathrm{negl}(\lambda)$$

Useful, e.g., for password hashing

Preimage resistance:

$$\Pr\left[H(s,x) = y \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^{\lambda} \\ y \leftarrow \{0,1\}^{\lambda} \\ x \leftarrow A(s,y) \end{array} \right] < \mathrm{negl}(\lambda)$$

$$f : \{0,1\}^n \to \{0,1\}^m$$

f is called a **one-way function** if for any PPT program A and for all inputs x the following probability is negligible (in n):

$$\Pr_{x \leftarrow \{0,1\}^n} \left[f(A(f(x)) = f(x) \right]$$

Collision Resistance

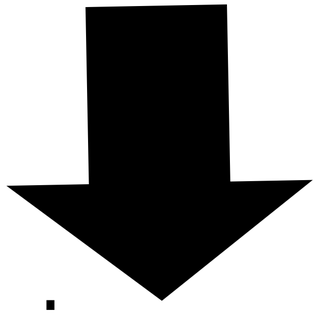
Consider a family of hash functions

$$H : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^\lambda$$

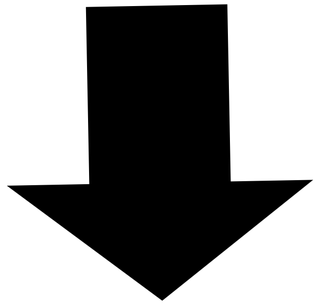
Important open theoretical problem: prove OWFs are insufficient to achieve collision resistance

$$\Pr \left[H(s, x_0) = H(s, x_1) \;\middle|\; \begin{array}{l} s \leftarrow \{0,1\}^\lambda \\ (x_0, x_1) \leftarrow A(s) \end{array} \right] < \text{negl}(\lambda)$$

Collision Resistance:

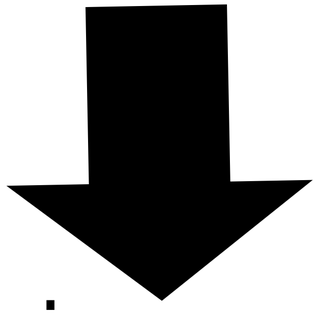


Second-preimage resistance:

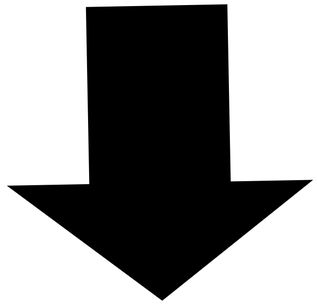


Preimage resistance:

Collision Resistance:



Second-preimage resistance:



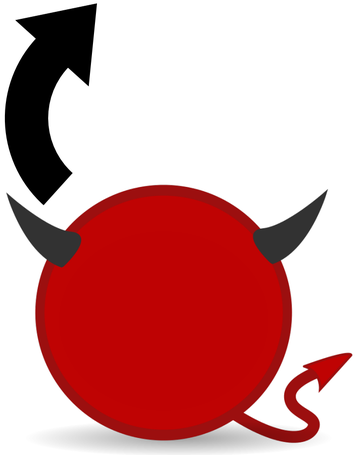
Preimage resistance:

All of these are trivial in random oracle model

Symmetric Key Cryptography

Alice

k



Eve

Bob

k

Confidentiality

Can Alice and Bob prevent Eve from listening?

Authenticity

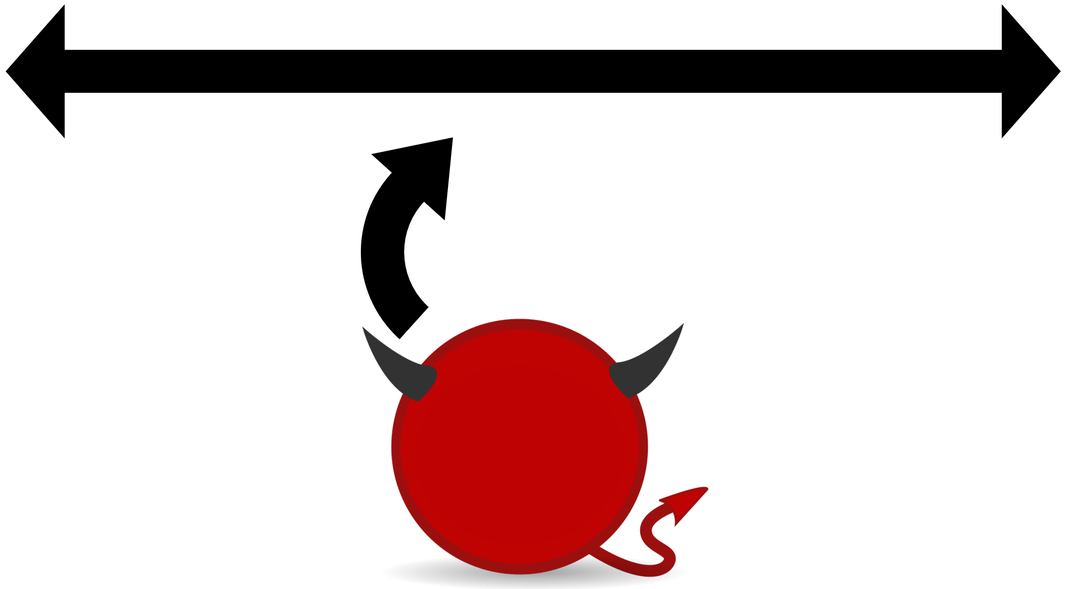
Can Bob be sure Eve did not send the message?

Can Bob be sure Eve did not alter a message from Alice?

Symmetric Key Cryptography

Alice

k



Bob

k

Eve

Concepts:

Kerckhoff’s Principle

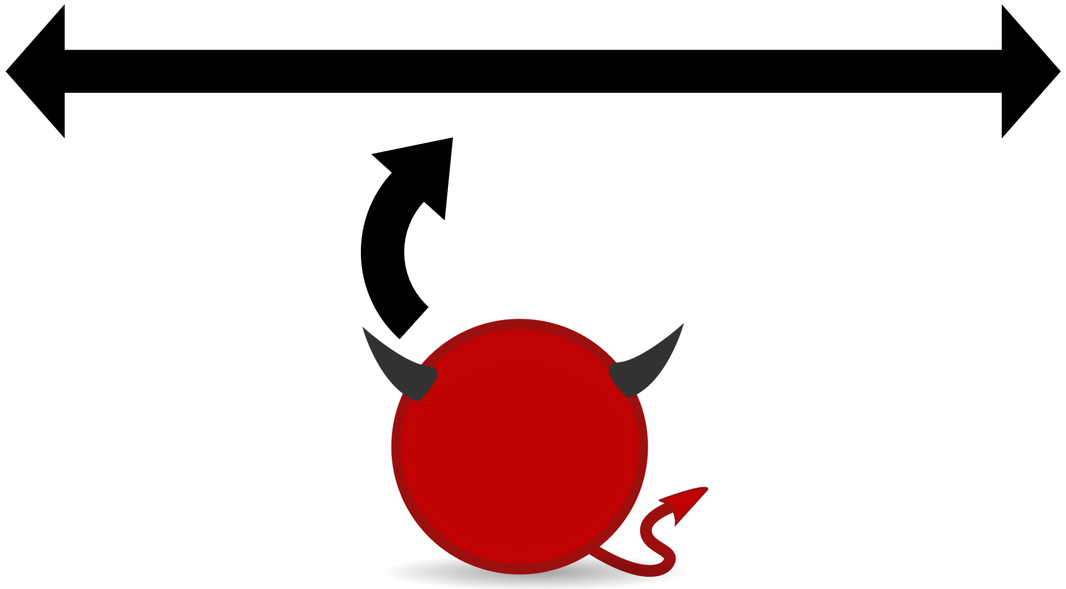
Shannon’s Theorem

Indistinguishability

Symmetric Key Cryptography

Alice

k



Bob

k

Eve

Concepts:

- Kerckhoff's Principle
- Shannon's Theorem
- Indistinguishability

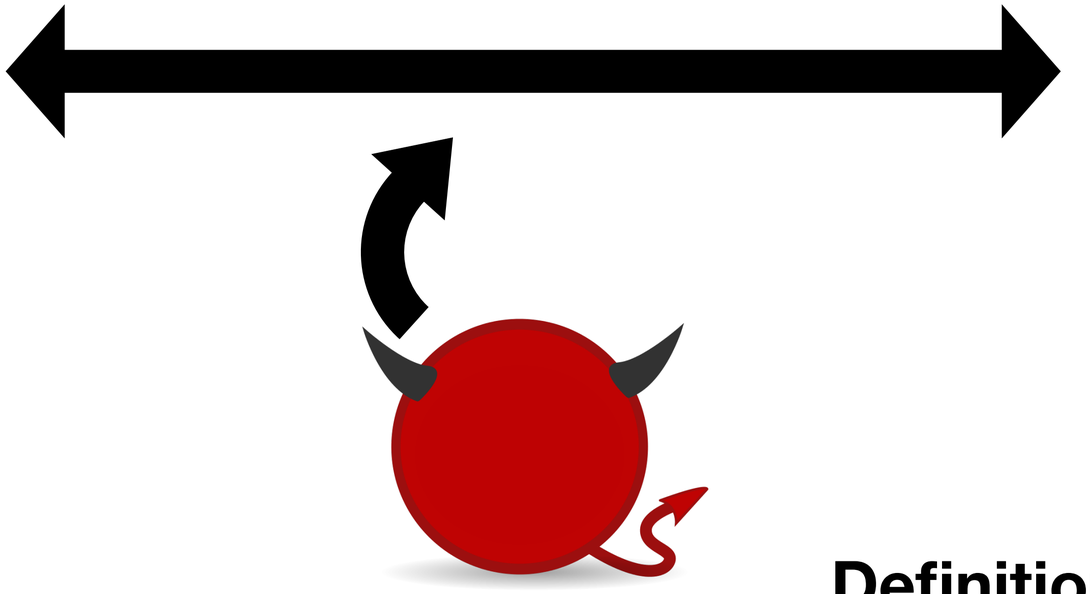
Tools:

- PRGs, PRFs, Block Ciphers, OWFs
- Hash Functions

Symmetric Key Cryptography

Alice

k



Bob

k

Eve

Concepts:

- Kerckhoff's Principle
- Shannon's Theorem
- Indistinguishability

Tools:

- PRGs, PRFs, Block Ciphers, OWFs
- Hash Functions

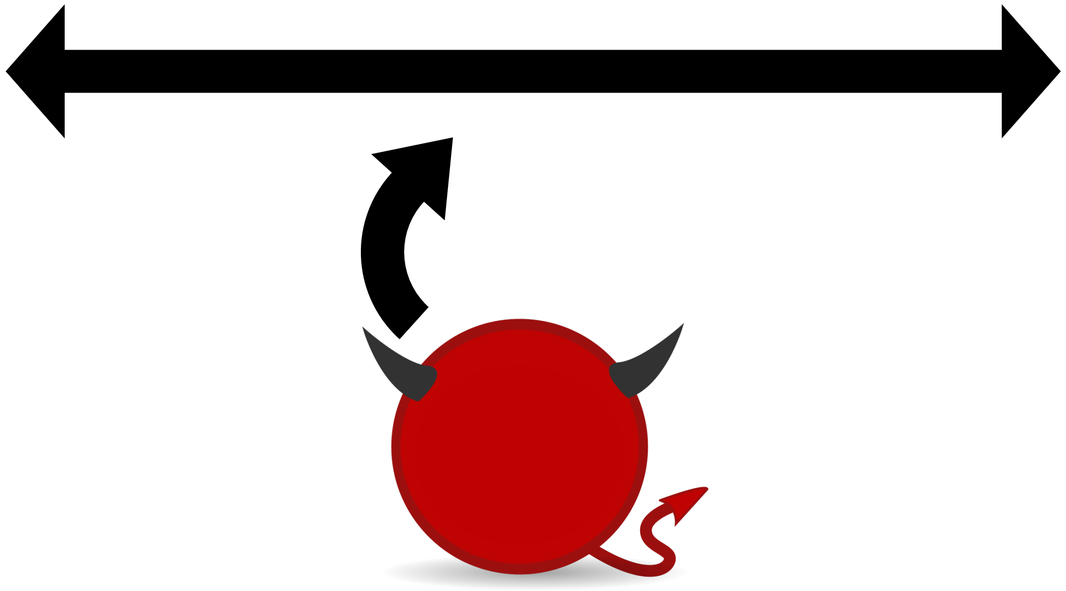
Definitions:

- CPA Security
- CCA Security
- MACs
- Authenticated Encryption/AE-AD

Symmetric Key Cryptography

Alice

k



Bob

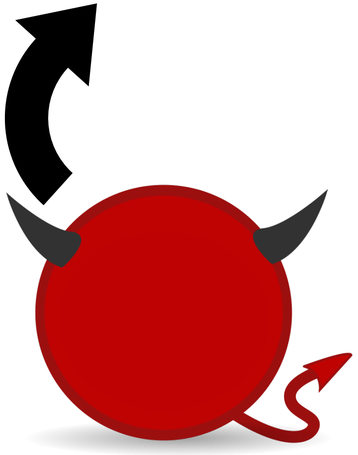
k

Eve

If Alice and Bob can exchange a short secret key, then they can use cryptography to construct a secure communication channel

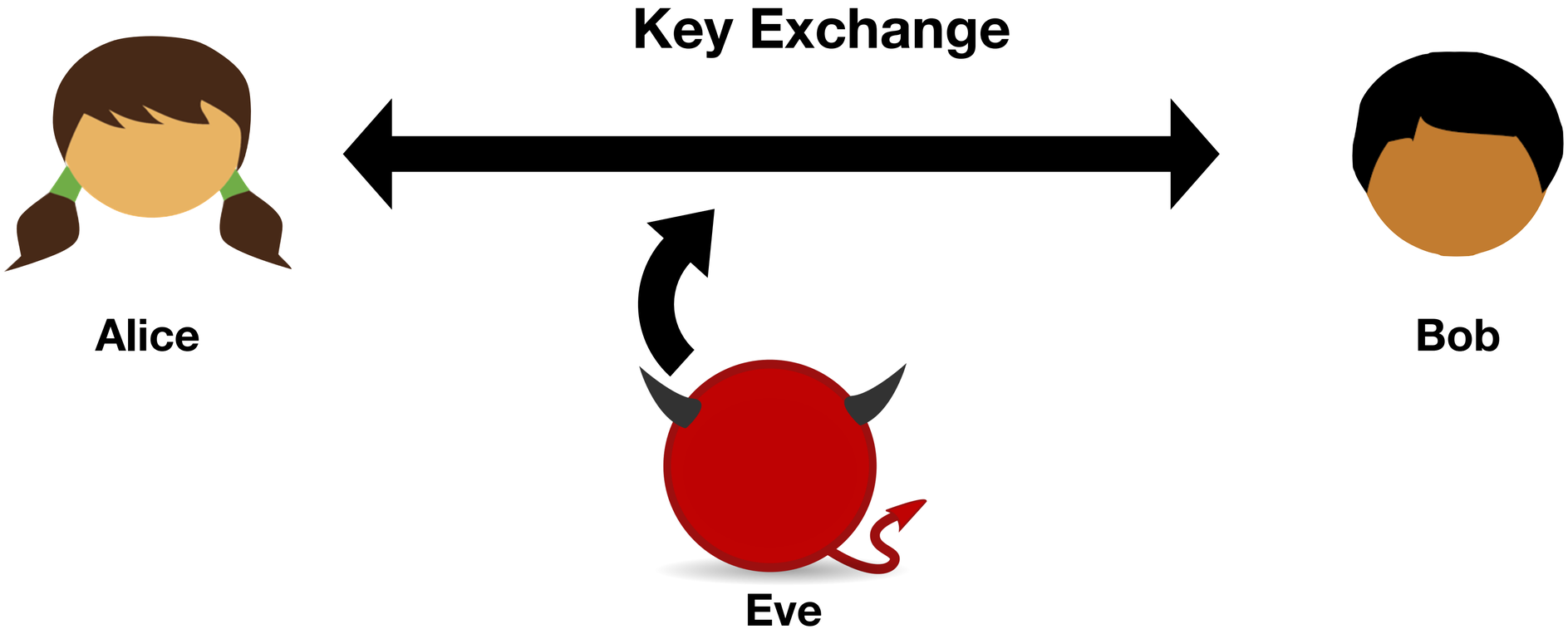
Key Exchange

Alice



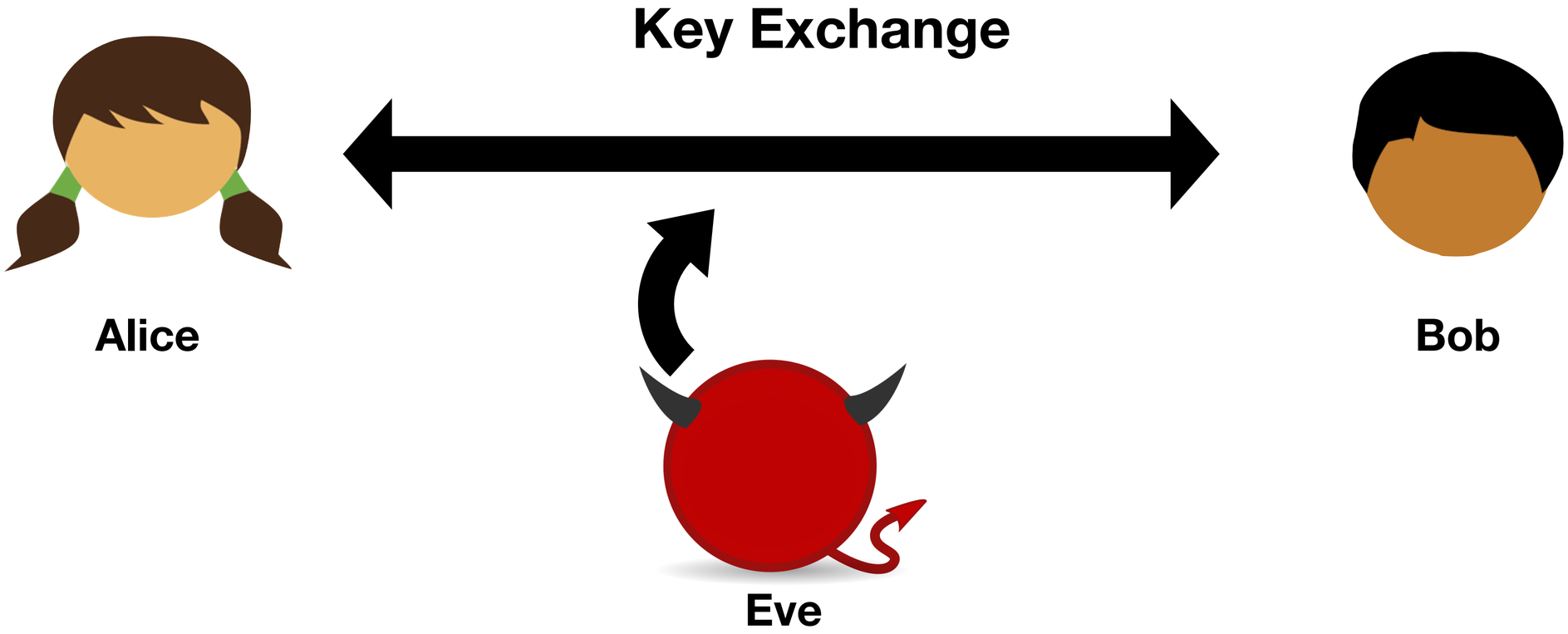
Eve

Bob



Alice and Bob communicate a few times, then agree on a secret key k

Under a cryptographic assumption, Eve does not know k



Alice and Bob communicate a few times, then agree on a secret key k

Under a cryptographic assumption, Eve does not know k

Important open theoretical problem: prove OWFs are insufficient to achieve key exchange

Limits on the Provable Consequences of One-way Permutations.

Russell Impagliazzo*
Computer Science Division
University of California at Berkeley
Berkeley, California 94720

Steven Rudich†
Computer Science Department
University of Toronto
Toronto, Canada M5S 1A4

March 9, 1989

Abstract

We present strong evidence that the implication, "if one-way permutations exist, then secure secret key agreement is possible", is not provable by standard techniques. Since both sides of this implication are widely believed true in real life, to show that the implication is false requires a new model. We consider a world where all parties have access to a black box for a randomly selected permutation. Being totally random, this permutation will be strongly one-way in a provable, information-theoretic way. We show that, if $P = NP$, no protocol for secret key agreement is secure in such a setting. Thus, to prove that a secret key agreement protocol which uses a one-way permutation as a black box is secure is as hard as proving $P \neq NP$. We also obtain, as a corollary, that there is an oracle relative to which the implication is false, i.e., there is a one-way permutation, yet secret-exchange is impossible. Thus, no technique which relativizes can prove that secret exchange can be based on any one-way permutation. Our results present a general framework for proving statements of the form, "Cryptographic application X is not likely possible based solely on complexity assumption Y."

*Research partially supported by NSF grant CCR 88-13632.

†Research partially supported by NSF grant CCR 88-13632 and an IBM doctoral fellowship.

1 Introduction.

A typical result in cryptography will be of the form: With assumption X, we can prove that a secure protocol for task P is possible. Because the standard cryptographic assumptions are, at present, unproved, many results focus on weakening the assumptions needed to imply that a given protocol is possible. As a consequence, we ask a new form of question: which assumptions are too weak to yield a proof that a secure protocol for P is possible?

The task we will study is secure secret-key agreement. Secret-key agreement is a protocol where Alice and Bob, having no secret information in common, agree on a secret-key over a public channel. Such a protocol is secure when no polynomial-time Eve listening to the conversation can determine part of the secret. Secure secret-key agreement is known to be possible under the assumption that trapdoor functions exist [DH76], [GM84]. However, researchers have been frustrated by unsuccessful attempts to base it on the weaker assumption that one-way permutations exist.

We provide strong evidence that it will be difficult to prove that secure secret-key agreement is possible assuming only that a one-way permutation exists. We model the existence of a one-way permutation by allowing all parties access to a randomly chosen permutation oracle. A random permutation oracle is provably one-way in the strongest possible sense. We show that any proof that secure secret-key agreement is possible in a world with a random permutation oracle would simultaneously prove $P \neq NP$. (Formally, $P = NP$ implies there is no secure secret-key agreement relative to a random permutation oracle.) We conclude that it is as

An Oracle Separation

Key exchange does not exist given only a random oracle

… But weirdly this does *not* mean that one-way functions cannot possibly imply key exchange

Limits on the Provable Consequences of One-way Permutations.

Russell Impagliazzo*
Computer Science Division
University of California at Berkeley
Berkeley, California 94720

Steven Rudich†
Computer Science Department
University of Toronto
Toronto, Canada M5S 1A4

March 9, 1989

Abstract

We present strong evidence that the implication, "if one-way permutations exist, then secure secret key agreement is possible", is not provable by standard techniques. Since both sides of this implication are widely believed true in real life, to show that the implication is false requires a new model. We consider a world where all parties have access to a black box for a randomly selected permutation. Being totally random, this permutation will be strongly one-way in a provable, information-theoretic way. We show that, if $P = NP$, no protocol for secret key agreement is secure in such a setting. Thus, to prove that a secret key agreement protocol which uses a one-way permutation as a black box is secure is as hard as proving $P \neq NP$. We also obtain, as a corollary, that there is an oracle relative to which the implication is false, i.e., there is a one-way permutation, yet secret-exchange is impossible. Thus, no technique which relativizes can prove that secret exchange can be based on any one-way permutation. Our results present a general framework for proving statements of the form, "Cryptographic application X is not likely possible based solely on complexity assumption Y."

*Research partially supported by NSF grant CCR 88-13632.

†Research partially supported by NSF grant CCR 88-13632 and an IBM doctoral fellowship.

1 Introduction.

A typical result in cryptography will be of the form: With assumption X, we can prove that a secure protocol for task P is possible. Because the standard cryptographic assumptions are, at present, unproved, many results focus on weakening the assumptions needed to imply that a given protocol is possible. As a consequence, we ask a new form of question: which assumptions are too weak to yield a proof that a secure protocol for P is possible?

The task we will study is secure secret-key agreement. Secret-key agreement is a protocol where Alice and Bob, having no secret information in common, agree on a secret-key over a public channel. Such a protocol is secure when no polynomial-time Eve listening to the conversation can determine part of the secret. Secure secret-key agreement is known to be possible under the assumption that trapdoor functions exist [DH76], [GM84]. However, researchers have been frustrated by unsuccessful attempts to base it on the weaker assumption that one-way permutations exist.

We provide strong evidence that it will be difficult to prove that secure secret-key agreement is possible assuming only that a one-way permutation exists. We model the existence of a one-way permutation by allowing all parties access to a randomly chosen permutation oracle. A random permutation oracle is provably one-way in the strongest possible sense. We show that any proof that secure secret-key agreement is possible in a world with a random permutation oracle would simultaneously prove $P \neq NP$. (Formally, $P = NP$ implies there is no secure secret-key agreement relative to a random permutation oracle.) We conclude that it is as

An Oracle Separation

Key exchange does not exist given only a random oracle

… But weirdly this does *not* mean that one-way functions cannot possibly imply key exchange

Might be possible to construct key exchange by using one-way functions in a non-black box way

Limits on the Provable Consequences of One-way Permutations.

Russell Impagliazzo*
Computer Science Division
University of California at Berkeley
Berkeley, California 94720

Steven Rudich†
Computer Science Department
University of Toronto
Toronto, Canada M5S 1A4

March 9, 1989

Abstract

We present strong evidence that the implication, "if one-way permutations exist, then secure secret key agreement is possible", is not provable by standard techniques. Since both sides of this implication are widely believed true in real life, to show that the implication is false requires a new model. We consider a world where all parties have access to a black box for a randomly selected permutation. Being totally random, this permutation will be strongly one-way in a provable, information-theoretic way. We show that, if $P = NP$, no protocol for secret key agreement is secure in such a setting. Thus, to prove that a secret key agreement protocol which uses a one-way permutation as a black box is secure is as hard as proving $P \neq NP$. We also obtain, as a corollary, that there is an oracle relative to which the implication is false, i.e., there is a one-way permutation, yet secret-exchange is impossible. Thus, no technique which relativizes can prove that secret exchange can be based on any one-way permutation. Our results present a general framework for proving statements of the form, "Cryptographic application X is not likely possible based solely on complexity assumption Y."

*Research partially supported by NSF grant CCR 88-13632.

†Research partially supported by NSF grant CCR 88-13632 and an IBM doctoral fellowship.

1 Introduction.

A typical result in cryptography will be of the form: With assumption X, we can prove that a secure protocol for task P is possible. Because the standard cryptographic assumptions are, at present, unproved, many results focus on weakening the assumptions needed to imply that a given protocol is possible. As a consequence, we ask a new form of question: which assumptions are too weak to yield a proof that a secure protocol for P is possible?

The task we will study is secure secret-key agreement. Secret-key agreement is a protocol where Alice and Bob, having no secret information in common, agree on a secret-key over a public channel. Such a protocol is secure when no polynomial-time Eve listening to the conversation can determine part of the secret. Secure secret-key agreement is known to be possible under the assumption that trapdoor functions exist [DH76], [GM84]. However, researchers have been frustrated by unsuccessful attempts to base it on the weaker assumption that one-way permutations exist.

We provide strong evidence that it will be difficult to prove that secure secret-key agreement is possible assuming only that a one-way permutation exists. We model the existence of a one-way permutation by allowing all parties access to a randomly chosen permutation oracle. A random permutation oracle is provably one-way in the strongest possible sense. We show that any proof that secure secret-key agreement is possible in a world with a random permutation oracle would simultaneously prove $P \neq NP$. (Formally, $P = NP$ implies there is no secure secret-key agreement relative to a random permutation oracle.) We conclude that it is as

An Oracle Separation

Key exchange does not exist given only a random oracle

… But weirdly this does *not* mean that one-way functions cannot possibly imply key exchange

Might be possible to construct key exchange by using one-way functions in a non-black box way

But most cryptographers think this is not possible

Today's objectives

Discuss concrete properties of hash functions

Define collision resistance and other concrete properties

Prove RO achieves these properties, discuss relationship between them

Review symmetric-key cryptography, and look forward to public-key cryptography