Limitations of Perfect Security

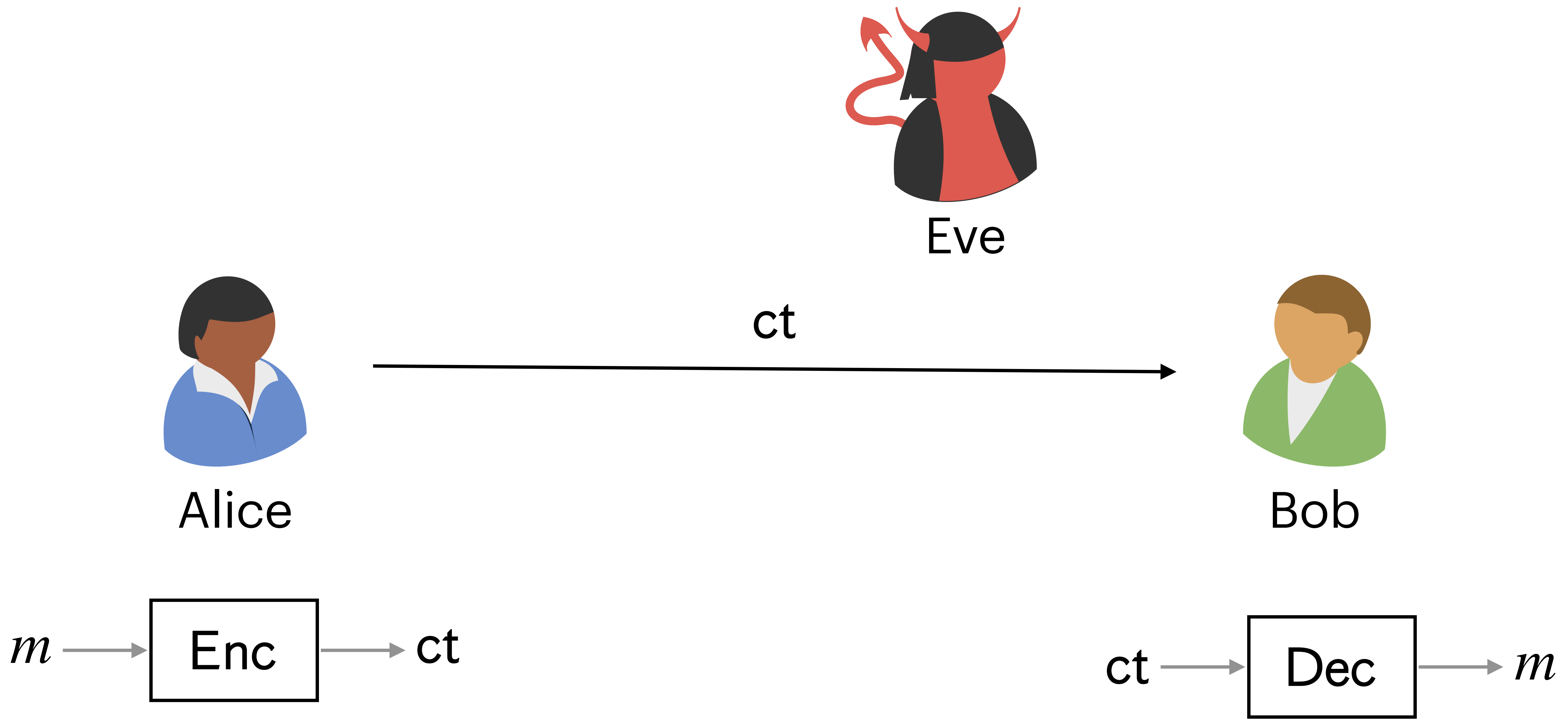
601.442/642 Modern Cryptography

27th January 2026

Announcement

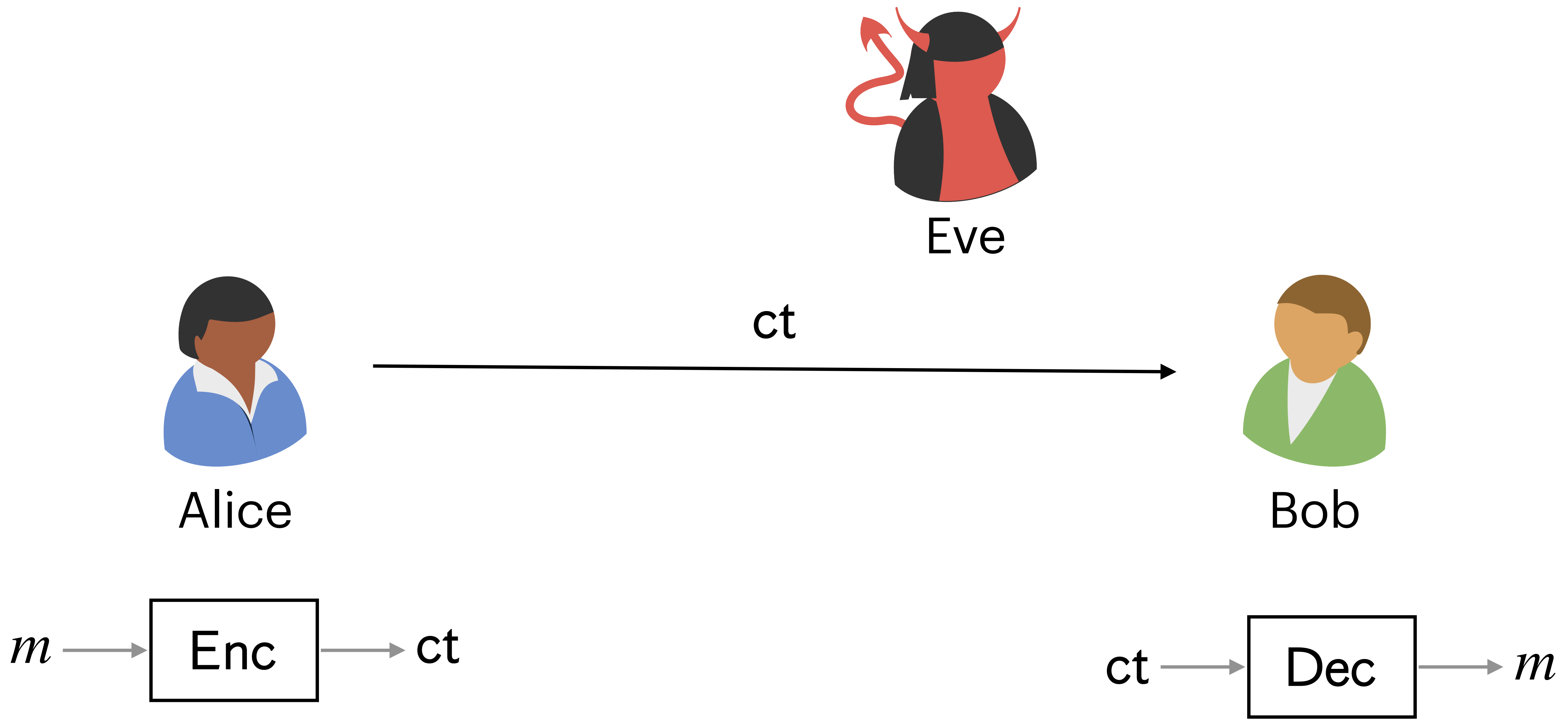
- Homework 1 due this **Thursday** (29th January)
- Please start early and come to office hours with any questions!

Recap: Encryption Scheme



Goal: Bob can decrypt ct but not Eve.

Recap: Encryption Scheme



Goal: Bob can decrypt **ct** but not Eve.

Kerckhoffs' Principle: The security of a cryptosystem shouldn't rely on the secrecy of the algorithm (only the key)

Recap: Encryption Scheme Syntax and Correctness

Encryption Scheme Syntax

An encryption scheme consists of three (possibly probabilistic) algorithms:

- $\mathsf{KeyGen}() \to k$ outputs a key $k \in \mathscr{K}$.
- $\mathsf{Enc}(k, m) \to \mathsf{ct}$ takes key k and message $m \in \mathscr{M}$ and outputs ciphertext $\mathsf{ct} \in \mathscr{C}$.
- $\mathsf{Dec}(k, \mathsf{ct}) \to m$ takes key k and ciphertext ct and outputs message m.

Recap: Encryption Scheme Syntax and Correctness

Encryption Scheme Syntax

An encryption scheme consists of three (possibly probabilistic) algorithms:

- $\mathsf{KeyGen}() \to k$ outputs a key $k \in \mathcal{K}$.
- $\mathsf{Enc}(k, m) \to \mathsf{ct}$ takes key k and message $m \in \mathcal{M}$ and outputs ciphertext $\mathsf{ct} \in \mathcal{C}$.
- $\mathsf{Dec}(k, \mathsf{ct}) \to m$ takes key k and ciphertext ct and outputs message m.

Encryption Scheme Correctness

An encryption scheme satisfies correctness if $\forall k \in \mathcal{K}, \forall m \in \mathcal{M}$, we have

$$\Pr[\mathsf{Dec}(k, \mathsf{Enc}(k, m)) = m] = 1,$$

where the probability is over the randomness used in encryption and decryption.

Recap: One-Time Uniform Ciphertext Security

- What the security definition should capture for encryption schemes like OTP
    - The secret key should be kept hidden from Eve.
    - The key is only used to encrypt one plaintext.
    - The ciphertext looks uniformly random to Eve.

Recap: One-Time Uniform Ciphertext Security

- What the security definition should capture for encryption schemes like OTP
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - The ciphertext looks uniformly random to Eve.

One-Time Uniform Ciphertext Security

An encryption scheme is one-time uniform ciphertext secure if $\forall m \in \mathscr{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{matrix} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m) \end{matrix} \right\} \quad \equiv \quad D_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathscr{C} \right\}$$

Recap: One-Time Uniform Ciphertext Security

- What the security definition should capture for encryption schemes like OTP
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - The ciphertext looks uniformly random to Eve.

One-Time Uniform Ciphertext Security

An encryption scheme is one-time uniform ciphertext secure if $\forall m \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m) \end{array} \right\} \equiv D_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathscr{C} \right\}$$

Eve's view when
Alice encrypts m

Recap: One-Time Uniform Ciphertext Security

- What the security definition should capture for encryption schemes like OTP
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - The ciphertext looks uniformly random to Eve.

One-Time Uniform Ciphertext Security

An encryption scheme is one-time uniform ciphertext secure if $\forall m \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m) \end{array} \right\} \equiv D_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathcal{C} \right\}$$

Eve's view when Alice encrypts m

What we want Eve's view to look like

Recap: One-Time Uniform Ciphertext Security

- What the security definition should capture for encryption schemes like OTP
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - The ciphertext looks uniformly random to Eve.

<u>One-Time Uniform Ciphertext Security</u>

An encryption scheme is one-time uniform ciphertext secure if $\forall m \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m) \end{array} \right\} \equiv D_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathscr{C} \right\}$$

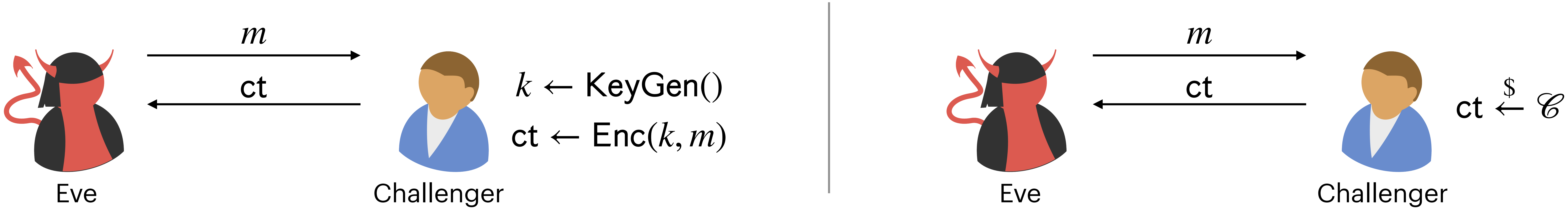
Eve's view when Alice encrypts m

What we want Eve's view to look like

Security: D_1 carries no information about the message

Alternative View of One-Time Uniform Ciphertext Security

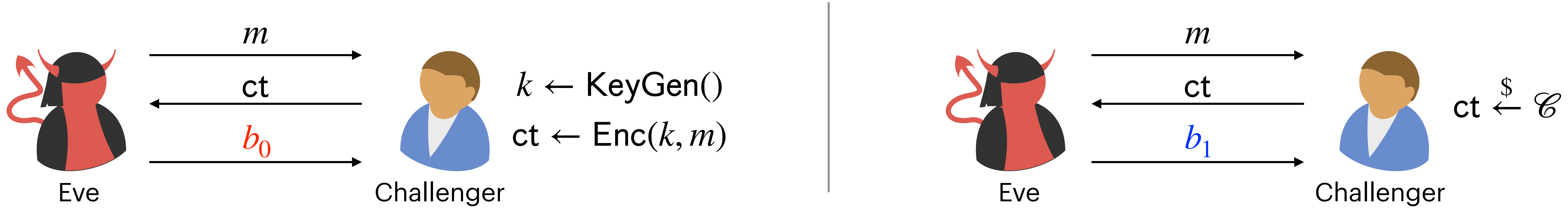
Consider the following two interactions between Eve and a challenger.



Encryption scheme is one-time uniform ciphertext secure if the above two scenarios seem identical to Eve.

Alternative View of One-Time Uniform Ciphertext Security

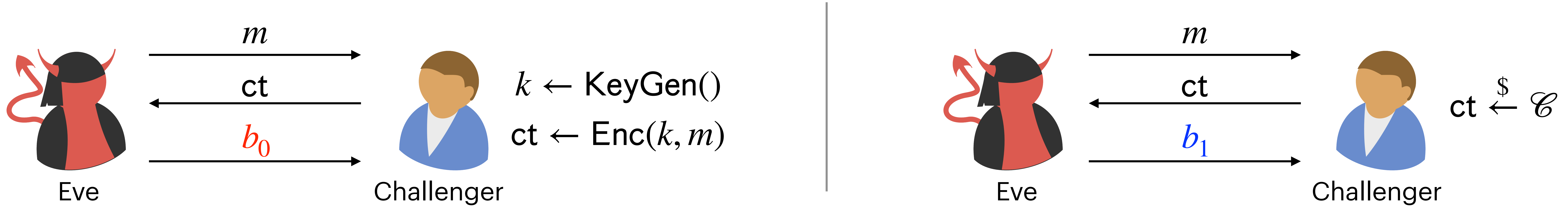
Consider the following two interactions between Eve and a challenger.



Encryption scheme is one-time uniform ciphertext secure if the above two scenarios seem identical to Eve.

Alternative View of One-Time Uniform Ciphertext Security

Consider the following two interactions between Eve and a challenger.



Encryption scheme is one-time uniform ciphertext secure if the above two scenarios seem identical to Eve.

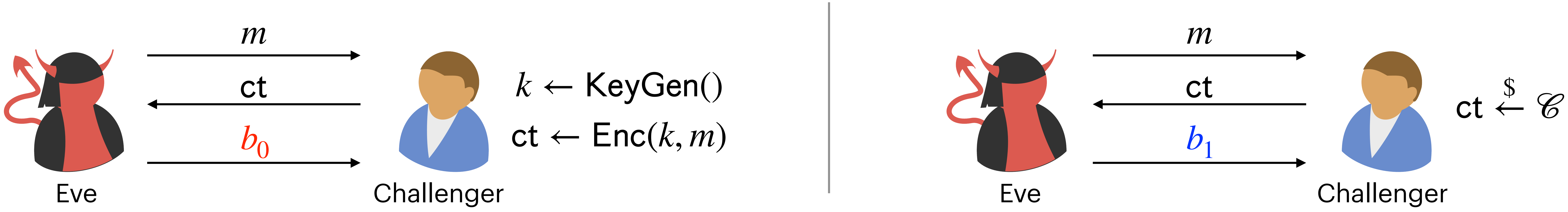
That is

$$\Pr[b_0 = 1] = \Pr[b_1 = 1]$$

where the probability is over the randomness of KeyGen and Enc.

Alternative View of One-Time Uniform Ciphertext Security

Consider the following two interactions between Eve and a challenger.



Encryption scheme is one-time uniform ciphertext secure if the above two scenarios seem identical to Eve.

That is

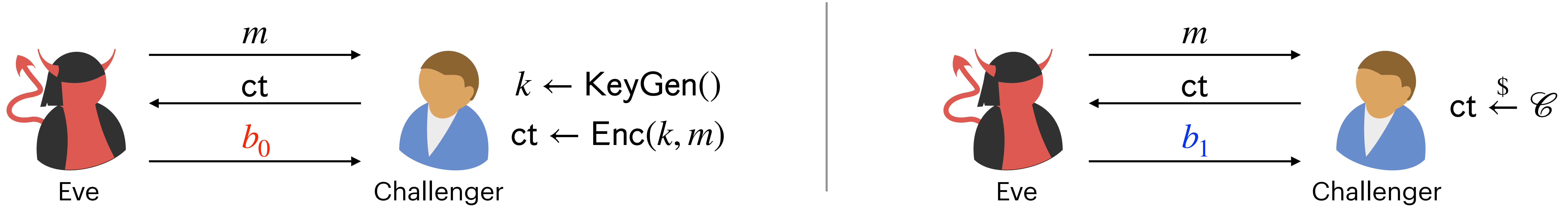
$$\Pr[b_0 = 1] = \Pr[b_1 = 1]$$

where the probability is over the randomness of KeyGen and Enc.

- Interaction with a challenger helps model what Eve can see during encryption and what remains hidden.

Alternative View of One-Time Uniform Ciphertext Security

Consider the following two interactions between Eve and a challenger.



Encryption scheme is one-time uniform ciphertext secure if the above two scenarios seem identical to Eve.

That is

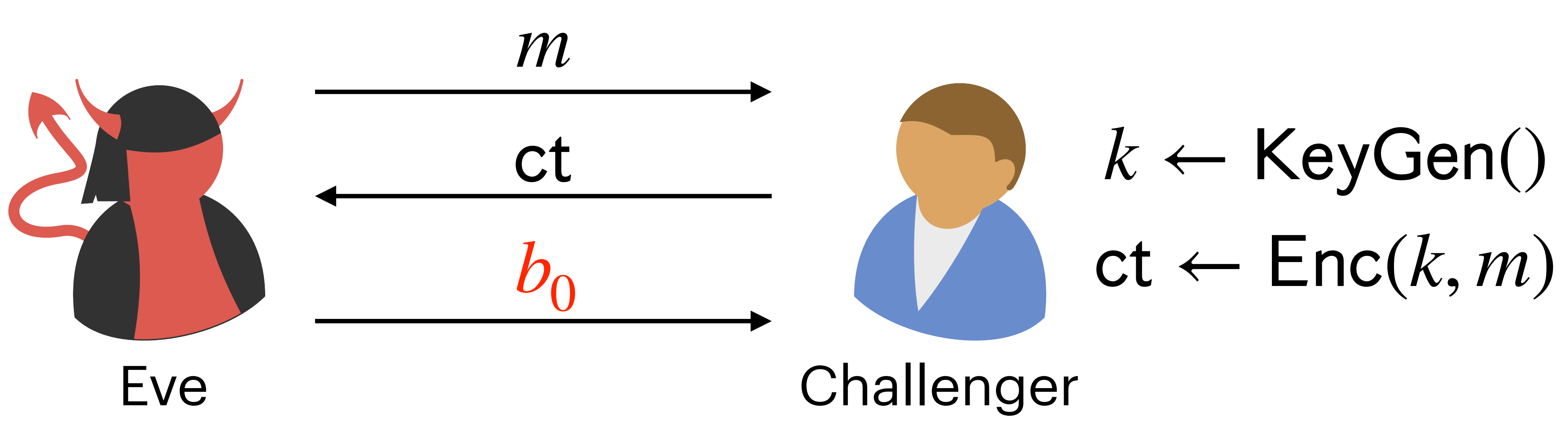
$$\Pr[b_0 = 1] = \Pr[b_1 = 1]$$

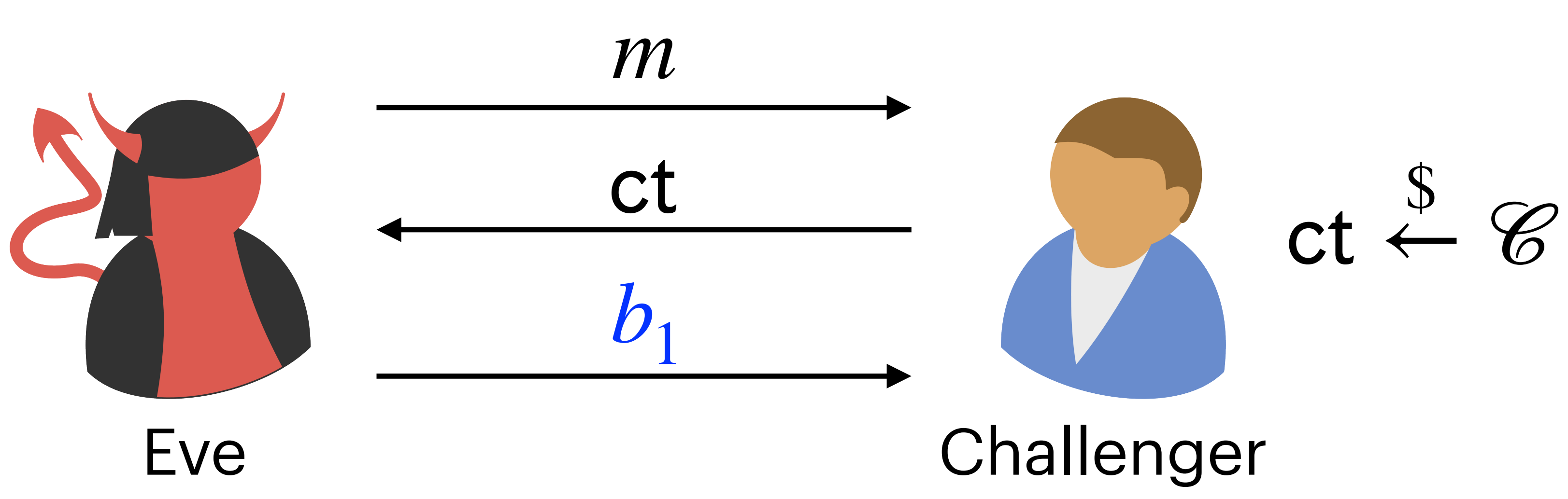
where the probability is over the randomness of KeyGen and Enc.

- Interaction with a challenger helps model what Eve can see during encryption and what remains hidden.
- Eve is allowed to choose the plaintext. If the scheme is secure when Eve chooses the plaintext, it is secure when she has only partial information about the plaintext.

Alternative View of One-Time Uniform Ciphertext Security

Consider the following two interactions between Eve and a challenger.





Encryption scheme is one-time uniform ciphertext secure if the above two scenarios seem identical to Eve.

That is

$$\Pr[b_0 = 1] = \Pr[b_1 = 1]$$

where the probability is over the randomness of KeyGen and Enc.

- Interaction with a challenger helps model what Eve can see during encryption and what remains hidden.
- Eve is allowed to choose the plaintext. If the scheme is secure when Eve chooses the plaintext, it is secure when she has only partial information about the plaintext.
- Equivalent to the previous definition of one-time uniform ciphertext security.

Recap: One-Time Pad

One-Time Pad

Let λ be a positive integer and let $\mathscr{K} = \mathscr{M} = \mathscr{C} = \{0,1\}^\lambda$.

- KeyGen(): $k \xleftarrow{\$} \{0,1\}^\lambda$.
- Enc(k, m): ct $:= k \oplus m$.
- Dec(k, ct): $m := k \oplus$ ct.

Recap: One-Time Pad

One-Time Pad

Let λ be a positive integer and let $\mathscr{K} = \mathscr{M} = \mathscr{C} = \{0,1\}^\lambda$.

- KeyGen(): $k \xleftarrow{\$} \{0,1\}^\lambda$.
- Enc(k, m): ct $:= k \oplus m$.
- Dec(k, ct): $m := k \oplus$ ct.

Theorem: One-time pad is correct and has one-time uniform ciphertext security.

Recap: Perfect Security

- An alternative idea for defining security of encryption schemes.
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - ~~The ciphertext looks uniformly random to Eve.~~ Encryptions of m_0 look like encryptions of m_1 to Eve.

(One-Time) Perfect Security

An encryption scheme is one-time perfectly secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \text{ct} : \begin{array}{c} k \leftarrow \text{KeyGen}() \\ \text{ct} \leftarrow \text{Enc}(k, m_0) \end{array} \right\} \equiv D_1 = \left\{ \text{ct} : \begin{array}{c} k \leftarrow \text{KeyGen}() \\ \text{ct} \leftarrow \text{Enc}(k, m_1) \end{array} \right\}$$

Recap: Perfect Security

- An alternative idea for defining security of encryption schemes.
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - ~~The ciphertext looks uniformly random to Eve.~~ Encryptions of m_0 look like encryptions of m_1 to Eve.

(One-Time) Perfect Security

An encryption scheme is one-time perfectly secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \text{ct} : \begin{array}{c} k \leftarrow \text{KeyGen}() \\ \text{ct} \leftarrow \text{Enc}(k, m_0) \end{array} \right\} \equiv D_1 = \left\{ \text{ct} : \begin{array}{c} k \leftarrow \text{KeyGen}() \\ \text{ct} \leftarrow \text{Enc}(k, m_1) \end{array} \right\}$$

Eve's view when
Alice encrypts m_0

Recap: Perfect Security

- An alternative idea for defining security of encryption schemes.
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - ~~The ciphertext looks uniformly random to Eve.~~ Encryptions of m_0 look like encryptions of m_1 to Eve.

(One-Time) Perfect Security

An encryption scheme is one-time perfectly secure if $\forall m_0, m_1 \in \mathscr{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \equiv D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}$$

Eve's view when Alice encrypts m_0

Eve's view should look like encryptions of m_1

Recap: Perfect Security

- An alternative idea for defining security of encryption schemes.
  - The secret key should be kept hidden from Eve.
  - The key is only used to encrypt one plaintext.
  - ~~The ciphertext looks uniformly random to Eve.~~ Encryptions of m_0 look like encryptions of m_1 to Eve.

(One-Time) Perfect Security

An encryption scheme is one-time perfectly secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \equiv D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}$$

Eve's view when Alice encrypts m_0

Eve's view should look like encryptions of m_1

Security: The ciphertext distribution is independent of the message.

Alternative View of Perfect Security

- Encryption scheme is perfectly secure if the above two scenarios seem identical to Eve.
  - Can be formalized similar to one-time uniform ciphertext security.

Alternative View of Perfect Security

- Encryption scheme is perfectly secure if the above two scenarios seem identical to Eve.
    - Can be formalized similar to one-time uniform ciphertext security.
- Interaction with a challenger helps model what Eve can see during encryption and what remains hidden.

Alternative View of Perfect Security

- Encryption scheme is perfectly secure if the above two scenarios seem identical to Eve.
  - Can be formalized similar to one-time uniform ciphertext security.
- Interaction with a challenger helps model what Eve can see during encryption and what remains hidden.
- Eve is allowed to choose the plaintexts.

Alternative View of Perfect Security

- Encryption scheme is perfectly secure if the above two scenarios seem identical to Eve.
  - Can be formalized similar to one-time uniform ciphertext security.
- Interaction with a challenger helps model what Eve can see during encryption and what remains hidden.
- Eve is allowed to choose the plaintexts.
- Equivalent to the previous definition of perfect security.

Recap: Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Recap: Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

$$H_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \quad \equiv \quad H_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathscr{C} \right\}$$

Recap: Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

$$H_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \quad \equiv \quad H_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathscr{C} \right\} \quad \equiv \quad H_2 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}$$

Recap: Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

$$H_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \equiv H_1 = \left\{ \mathsf{ct} : \mathsf{ct} \xleftarrow{\$} \mathscr{C} \right\} \equiv H_2 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}$$

Corollary: One-time pad is perfectly secure.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^{\lambda}$

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^\lambda$
  - Enc(k, m): Compute ct$' := k \oplus m$, sample $b_0, b_1 \leftarrow$ Ber$(3/4)$ and output ct := ct$||b_0 b_1$.

$b = 1$ with probability $3/4$ and is 0 otherwise

Concatenate two strings

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^\lambda$
  - Enc(k, m): Compute ct$' := k \oplus m$, sample $b_0, b_1 \leftarrow$ Ber$(3/4)$ and output ct := ct$||b_0 b_1$.
  - Dec$(k,$ ct$)$: Compute ct$'$ = ct$[0 : \lambda]$ and output $m = k \oplus$ ct$'$.

$b = 1$ with probability $3/4$ and is 0 otherwise

Concatenate two strings

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^\lambda$
  - Enc(k, m): Compute ct$' := k \oplus m$, sample $b_0, b_1 \leftarrow$ Ber$(3/4)$ and output ct $:=$ ct$||b_0 b_1$.
  - Dec$(k,$ ct$)$: Compute ct$' =$ ct$[0 : \lambda]$ and output $m = k \oplus$ ct$'$.

$b = 1$ with probability $3/4$ and is 0 otherwise

Concatenate two strings

Is this scheme one-time uniform ciphertext secure?

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^\lambda$
  - Enc(k, m): Compute ct$' := k \oplus m$, sample $b_0, b_1 \leftarrow$ Ber$(3/4)$ and output ct := ct$||b_0 b_1$.
  - Dec$(k,$ ct$)$: Compute ct$'$ = ct$[0 : \lambda]$ and output $m = k \oplus$ ct$'$.

$b = 1$ with probability $3/4$ and is 0 otherwise

Concatenate two strings

Is this scheme one-time uniform ciphertext secure? ✗

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^\lambda$
  - Enc(k, m): Compute $\mathsf{ct}' := k \oplus m$, sample $b_0, b_1 \leftarrow \mathsf{Ber}(3/4)$ and output $\mathsf{ct} := \mathsf{ct}||b_0 b_1$.
  - Dec(k, ct): Compute $\mathsf{ct}' = \mathsf{ct}[0:\lambda]$ and output $m = k \oplus \mathsf{ct}'$.

$b = 1$ with probability $3/4$ and is 0 otherwise

Concatenate two strings

Is this scheme one-time uniform ciphertext secure? ✗

Is this scheme perfectly secure?

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Proof:

- In other words, there exists an encryption scheme that is perfectly secure but not one-time uniform ciphertext secure.
- Consider the following encryption scheme:
  - KeyGen(): $k \leftarrow \{0,1\}^\lambda$
  - Enc(k, m): Compute ct$' := k \oplus m$, sample $b_0, b_1 \leftarrow$ Ber$(3/4)$ and output ct $:=$ ct$||b_0 b_1$.
  - Dec$(k,$ ct$)$: Compute ct$' =$ ct$[0 : \lambda]$ and output $m = k \oplus$ ct$'$.

$b = 1$ with probability $3/4$ and is 0 otherwise

Concatenate two strings

Is this scheme one-time uniform ciphertext secure? ✗

Is this scheme perfectly secure? ✓

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Therefore, perfect security is weaker than one-time uniform ciphertext security.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Therefore, perfect security is weaker than one-time uniform ciphertext security.

Security (Intuitive): The ciphertext does not reveal any information about the plaintext to Eve, no matter what she does with the ciphertext.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Therefore, perfect security is weaker than one-time uniform ciphertext security.

Security (Intuitive): The ciphertext does not reveal any information about the plaintext to Eve, no matter what she does with the ciphertext.

One-time uniform ciphertext security might be too strong.

Comparing Both Security Notions

Claim: If an encryption scheme is one-time uniform ciphertext secure, then it is also perfectly secure.

Claim: Perfect security does not necessarily imply one-time uniform ciphertext security.

Therefore, perfect security is weaker than one-time uniform ciphertext security.

Security (Intuitive): The ciphertext does not reveal any information about the plaintext to Eve, no matter what she does with the ciphertext.

One-time uniform ciphertext security might be too strong.

Perfect security exactly captures our intuition.

Limitations of Perfect Security

- Limitations of one-time pad:

Limitations of Perfect Security

- Limitations of one-time pad:
    - Key is as long as the message.

Limitations of Perfect Security

- Limitations of one-time pad:
  - Key is as long as the message.
  - A key cannot be used to encrypt more than one plaintext.

Limitations of Perfect Security

- Limitations of one-time pad:
  - Key is as long as the message.
  - A key cannot be used to encrypt more than one plaintext.
- These limitations hold for any perfectly secure encryption scheme!

Limitations of Perfect Security

- Limitations of one-time pad:
  - Key is as long as the message.
  - A key cannot be used to encrypt more than one plaintext.
- These limitations hold for any perfectly secure encryption scheme!

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Limitations of Perfect Security

- Limitations of one-time pad:
  - Key is as long as the message.
  - A key cannot be used to encrypt more than one plaintext.
- These limitations hold for any perfectly secure encryption scheme!

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

When $\mathscr{K}$ and $\mathscr{M}$ consist of fixed length strings $\Longrightarrow$ key is as long as the message.

Limitations of Perfect Security

- Limitations of one-time pad:
  - Key is as long as the message.
  - A key cannot be used to encrypt more than one plaintext.
- These limitations hold for any perfectly secure encryption scheme!

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathcal{K}$ and message space $\mathcal{M}$ satisfies

$$|\mathcal{K}| \geq |\mathcal{M}|.$$

When $\mathcal{K}$ and $\mathcal{M}$ consist of fixed length strings $\Longrightarrow$ key is as long as the message.

Extends immediately to t-message perfect security: $|\mathcal{K}| \geq |\mathcal{M}|^t$.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

- Consider some ciphertext $\mathsf{ct} \in \mathscr{C}$.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

- Consider some ciphertext $\mathsf{ct} \in \mathscr{C}$.
- Every $m \in \mathscr{M}$ should be a valid decryption of ct under some key.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

- Consider some ciphertext ct $\in \mathscr{C}$.
- Every $m \in \mathscr{M}$ should be a valid decryption of ct under some key.
  - Follows from perfect security: Requires that ct cannot rule out any message.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

- Consider some ciphertext ct $\in \mathscr{C}$.
- Every $m \in \mathscr{M}$ should be a valid decryption of ct under some key.
  - Follows from perfect security: Requires that ct cannot rule out any message.
- A single key $k \in \mathscr{K}$ cannot decrypt ct to two different messages.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

- Consider some ciphertext $\mathsf{ct} \in \mathscr{C}$.
- Every $m \in \mathscr{M}$ should be a valid decryption of ct under some key.
  - Follows from perfect security: Requires that ct cannot rule out any message.
- A single key $k \in \mathscr{K}$ cannot decrypt ct to two different messages.
  - Follows from correctness: For each $m \in \mathscr{M}$, the key k that decrypts ct to m must be distinct.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Intuition:

- Consider some ciphertext ct $\in \mathscr{C}$.
- Every $m \in \mathscr{M}$ should be a valid decryption of ct under some key.
  - Follows from perfect security: Requires that ct cannot rule out any message.
- A single key $k \in \mathscr{K}$ cannot decrypt ct to two different messages.
  - Follows from correctness: For each $m \in \mathscr{M}$, the key k that decrypts ct to m must be distinct.
- Since there are $|\mathscr{M}|$ messages, there must be at least $|\mathscr{M}|$ keys. Thus, $|\mathscr{K}| \geq |\mathscr{M}|$.

Limitations of Perfect Security

Theorem (Shannon): Any perfectly secure encryption scheme with key space $\mathscr{K}$ and message space $\mathscr{M}$ satisfies

$$|\mathscr{K}| \geq |\mathscr{M}|.$$

Perfect security is too strong. Can we weaken the definition?

Encryption: Statistical Security

- Statistical security requires that the distribution of encryptions of m_0 is "close" to the distribution of encryptions of m_1.

Encryption: Statistical Security

- Statistical security requires that the distribution of encryptions of m_0 is "close" to the distribution of encryptions of m_1.

(One-Time) Statistical Security

An encryption scheme is one-time ϵ-statistically secure if $\forall m_0, m_1 \in \mathscr{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \overset{\epsilon}{\approx} D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}.$$

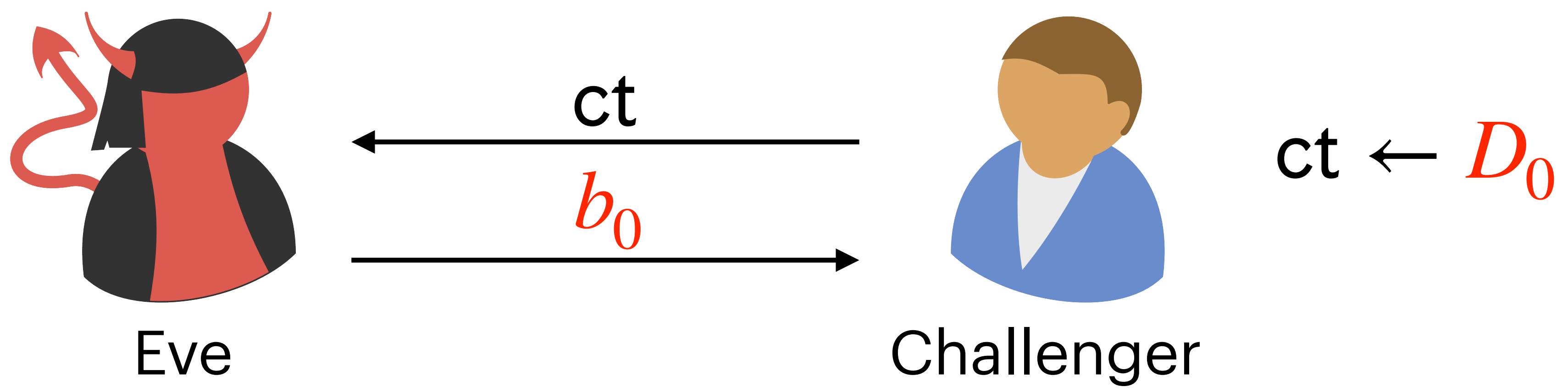
Encryption: Statistical Security

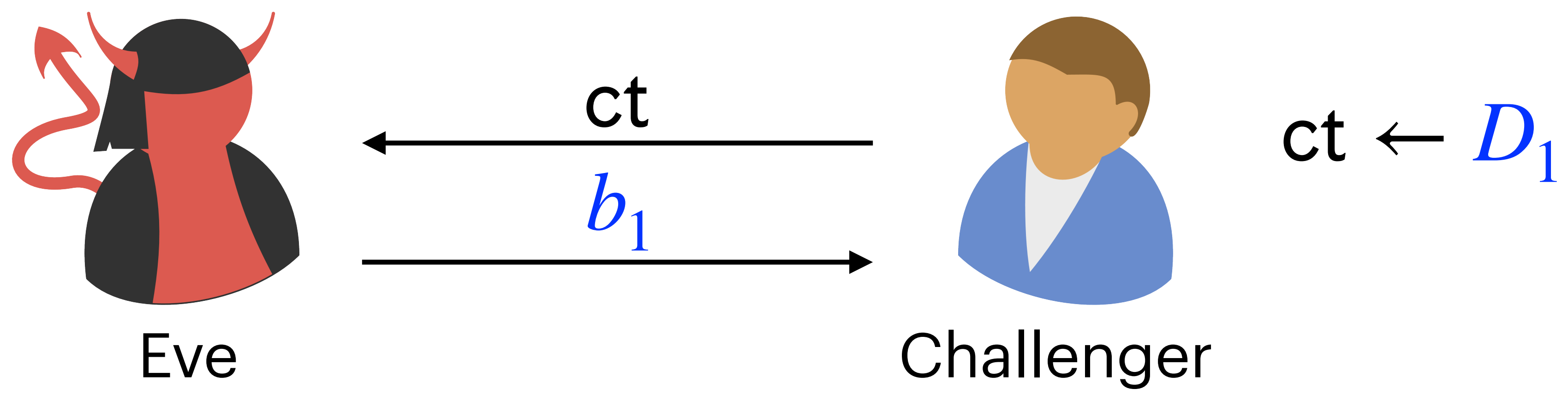
- Statistical security requires that the distribution of encryptions of m_0 is "close" to the distribution of encryptions of m_1.

(One-Time) Statistical Security

An encryption scheme is one-time ϵ-statistically secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \overset{\epsilon}{\approx} D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}.$$





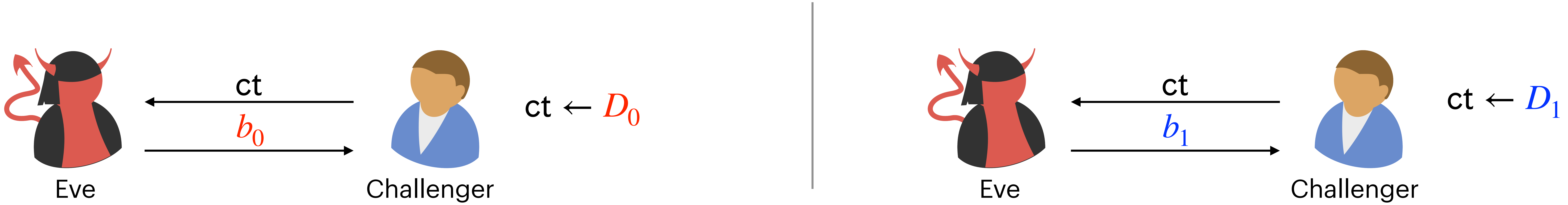
Encryption: Statistical Security

- Statistical security requires that the distribution of encryptions of m_0 is "close" to the distribution of encryptions of m_1.

(One-Time) Statistical Security

An encryption scheme is one-time ϵ-statistically secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \quad \overset{\epsilon}{\approx} \quad D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}.$$



$D_0 \overset{\epsilon}{\approx} D_1$ if

$$|\Pr[b_0 = 1] - \Pr[b_1 = 1]| \leq \epsilon$$

where the probability is over the randomness of KeyGen and Enc.

Encryption: Statistical Security

- Statistical security requires that the distribution of encryptions of m_0 is "close" to the distribution of encryptions of m_1.

(One-Time) Statistical Security

An encryption scheme is one-time ϵ-statistically secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \overset{\epsilon}{\approx} D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}.$$

Encryption: Statistical Security

- Statistical security requires that the distribution of encryptions of m_0 is "close" to the distribution of encryptions of m_1.

(One-Time) Statistical Security

An encryption scheme is one-time ϵ-statistically secure if $\forall m_0, m_1 \in \mathcal{M}$,

$$D_0 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_0) \end{array} \right\} \overset{\epsilon}{\approx} D_1 = \left\{ \mathsf{ct} : \begin{array}{c} k \leftarrow \mathsf{KeyGen}() \\ \mathsf{ct} \leftarrow \mathsf{Enc}(k, m_1) \end{array} \right\}.$$

Shannon's theorem can be extended to show that statistically secure encryption schemes still require long keys.

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?

- Consider an Eve that tries all possible keys for one-time pad

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?

- Consider an Eve that tries all possible keys for one-time pad

  - **Example:** Let $\lambda = 3$ and let Eve obtain a ciphertext $\mathsf{ct} = 010$.

Pr	k	$m = k \oplus 010$
1/8	000	010
1/8	001	011
1/8	010	000
1/8	011	001
1/8	100	110
1/8	101	111
1/8	110	100
1/8	111	101

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?
- Consider an Eve that tries all possible keys for one-time pad
  - **Example:** Let $\lambda = 3$ and let Eve obtain a ciphertext $\mathsf{ct} = 010$.
- Eve does not learn anything about the message even if she carries out a brute force attack.

Pr	k	$m = k \oplus 010$
$1/8$	000	010
$1/8$	001	011
$1/8$	010	000
$1/8$	011	001
$1/8$	100	110
$1/8$	101	111
$1/8$	110	100
$1/8$	111	101

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?

- Consider an Eve that tries all possible keys for one-time pad

    - **Example:** Let $\lambda = 3$ and let Eve obtain a ciphertext $\mathsf{ct} = 010$.

- Eve does not learn anything about the message even if she carries out a brute force attack.

    - How much computation is needed for a brute force attack on λ-bit keys?

Pr	k	$m = k \oplus 010$
1/8	000	010
1/8	001	011
1/8	010	000
1/8	011	001
1/8	100	110
1/8	101	111
1/8	110	100
1/8	111	101

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?

- Consider an Eve that tries all possible keys for one-time pad

  - **Example:** Let $\lambda = 3$ and let Eve obtain a ciphertext $\mathsf{ct} = 010$.

- Eve does not learn anything about the message even if she carries out a brute force attack.

  - How much computation is needed for a brute force attack on λ-bit keys? 2^λ

Pr	k	$m = k \oplus 010$
1/8	000	010
1/8	001	011
1/8	010	000
1/8	011	001
1/8	100	110
1/8	101	111
1/8	110	100
1/8	111	101

Limitations of Perfect and Statistical Security

- Both perfect and statistical security seem too strong. Can we further weaken the definition?

- Consider an Eve that tries all possible keys for one-time pad

  - **Example:** Let $\lambda = 3$ and let Eve obtain a ciphertext $\mathsf{ct} = 010$.

- Eve does not learn anything about the message even if she carries out a brute force attack.

  - How much computation is needed for a brute force attack on λ-bit keys? 2^λ

- What if we relax security to only hold against attacks that are feasible to carry out?

Pr	k	$m = k \oplus 010$
1/8	000	010
1/8	001	011
1/8	010	000
1/8	011	001
1/8	100	110
1/8	101	111
1/8	110	100
1/8	111	101