

One-time pad security:

OTP:

$$\mathcal{K} = \{0, 1\}^\lambda$$

$$\mathcal{M} = \{0, 1\}^\lambda$$

$$\mathcal{C} = \{0, 1\}^\lambda$$

KeyGen:

$$k \leftarrow \mathcal{K}$$

return k

Enc(k, m):

$$\text{return } k \oplus m$$

Dec(k, c):

$$\text{return } k \oplus c$$

Claim:

OTP satisfies one-time secrecy. That is, $\mathcal{L}_{\text{ots-L}}^{\text{OTP}} \equiv \mathcal{L}_{\text{ots-R}}^{\text{OTP}}$.

We will **use** the fact that OTP ciphertexts are uniformly distributed:

$$\frac{\text{CTXT}(m \in \{0, 1\}^\lambda):}{k \leftarrow \{0, 1\}^\lambda \text{ return } k \oplus m} \equiv \frac{\text{CTXT}(m \in \{0, 1\}^\lambda):}{c \leftarrow \{0, 1\}^\lambda \text{ return } c}$$

Overview:

Want to show:

$\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$		$\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$
<u>QUERY($m_L, m_R \in \text{OTP}.\mathcal{M}$):</u> $k \leftarrow \text{OTP.KeyGen}$ $c := \text{OTP.Enc}(k, m_L)$ return c	\equiv	<u>QUERY($m_L, m_R \in \text{OTP}.\mathcal{M}$):</u> $k \leftarrow \text{OTP.KeyGen}$ $c := \text{OTP.Enc}(k, m_R)$ return c

Standard hybrid technique:

- ▶ Starting with $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$, make a sequence of small modifications
- ▶ Each modification has no effect on calling program / adversary
- ▶ Sequence of modifications ends with $\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$

Security proof


$$\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$$

QUERY($m_L, m_R \in \text{OTP}.\mathcal{M}$):

$k \leftarrow \text{OTP.KeyGen}$

$c := \text{OTP.Enc}(k, m_L)$

return c

Starting point is $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$.

Security proof

 $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$

QUERY($m_L, m_R \in \text{OTP}.\mathcal{M}$):

$k \leftarrow \text{OTP.KeyGen}$

$c := \text{OTP.Enc}(k, m_L)$

return c

Starting point is $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$. Fill in details of OTP

Security proof



$\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$	
QUERY($m_L, m_R \in \{0, 1\}^\lambda$):	
<hr/>	
$k \leftarrow \{0, 1\}^\lambda$	
$c := k \oplus m_L$	
return c	

Starting point is $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$. Fill in details of OTP

Security proof



$\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$

QUERY($m_L, m_R \in \{0, 1\}^\lambda$):

$k \leftarrow \{0, 1\}^\lambda$

$c := k \oplus m_L$

return c

Starting point is $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$. Fill in details of OTP

Security proof



$\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$

QUERY($m_L, m_R \in \{0, 1\}^\lambda$):

$k \leftarrow \{0, 1\}^\lambda$

$c := k \oplus m_L$

return c

These statements appear also in $\mathcal{L}_{\text{otp-real}}$.

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_L)$
return c

◇

$\mathcal{L}_{\text{otp-real}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$k \leftarrow \{0, 1\}^\lambda$
return $k \oplus m$

Factor out so that $\mathcal{L}_{\text{otp-real}}$ appears.

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_L)$
return c

◇

$\mathcal{L}_{\text{otp-real}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$k \leftarrow \{0, 1\}^\lambda$
return $k \oplus m$

Factor out so that $\mathcal{L}_{\text{otp-real}}$ appears.

Security proof



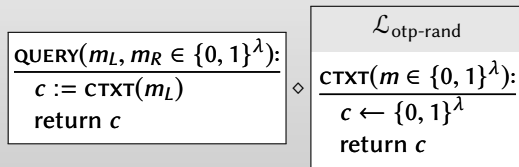
$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_L)$
return c



$\mathcal{L}_{\text{otp-rand}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$c \leftarrow \{0, 1\}^\lambda$
return c

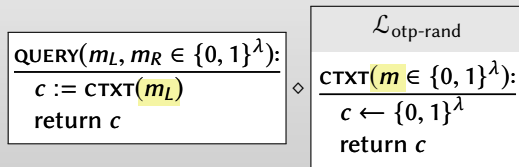
$\mathcal{L}_{\text{otp-real}}$ can be replaced with $\mathcal{L}_{\text{otp-rand}}$.

Security proof



$\mathcal{L}_{\text{otp-real}}$ can be replaced with $\mathcal{L}_{\text{otp-rand}}$.

Security proof



Argument to CTXT is never used!

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_R)$
return c



$\mathcal{L}_{\text{otp-rand}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$c \leftarrow \{0, 1\}^\lambda$
return c

Unused argument can be changed to m_R .

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_R)$
return c



$\mathcal{L}_{\text{otp-rand}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$c \leftarrow \{0, 1\}^\lambda$
return c

Unused argument can be changed to m_R .

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_R)$
return c



$\mathcal{L}_{\text{otp-real}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$k \leftarrow \{0, 1\}^\lambda$
return $k \oplus m$

$\mathcal{L}_{\text{otp-rand}}$ can be replaced with $\mathcal{L}_{\text{otp-real}}$.

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_R)$
return c

◇

$\mathcal{L}_{\text{otp-real}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$k \leftarrow \{0, 1\}^\lambda$
return $k \oplus m$

$\mathcal{L}_{\text{otp-rand}}$ can be replaced with $\mathcal{L}_{\text{otp-real}}$.

Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^\lambda):$
$c := \text{CTXT}(m_R)$
return c



$\mathcal{L}_{\text{otp-real}}$
$\text{CTXT}(m \in \{0, 1\}^\lambda):$
$k \leftarrow \{0, 1\}^\lambda$
return $k \oplus m$

Inline the subroutine call.

Security proof



QUERY($m_L, m_R \in \{0, 1\}^\lambda$):

$k \leftarrow \{0, 1\}^\lambda$

$c := k \oplus m_R$

return c

Inline the subroutine call.

Security proof



QUERY($m_L, m_R \in \{0, 1\}^\lambda$):

$k \leftarrow \{0, 1\}^\lambda$

$c := k \oplus m_R$

return c

Inline the subroutine call.

Security proof



QUERY($m_L, m_R \in \{0, 1\}^\lambda$):

$k \leftarrow \{0, 1\}^\lambda$

$c := k \oplus m_R$

return c

This happens to be $\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$.

Security proof ● ● ● ● ● ● ● ●

$\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$

QUERY($m_L, m_R \in \text{OTP}.\mathcal{M}$):

$k \leftarrow \text{OTP.KeyGen}$

$c := \text{OTP.Enc}(k, m_R)$

return c

This happens to be $\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$.

Security proof


$$\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$$

QUERY($m_L, m_R \in \text{OTP}.\mathcal{M}$):

$k \leftarrow \text{OTP.KeyGen}$

$c := \text{OTP.Enc}(k, m_R)$

return c

This happens to be $\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$.