

## Pseudo-one-time pad security:

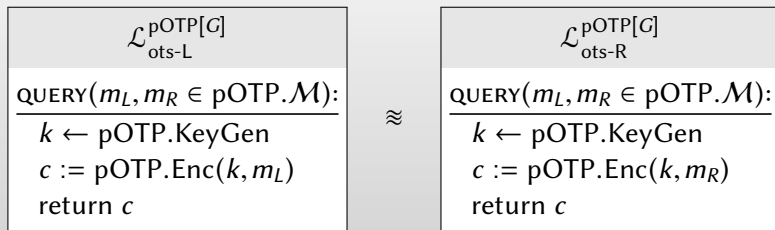
pOTP[G]:			
$\mathcal{K} = \{0, 1\}^\lambda$	$\text{KeyGen:}$	$\text{Enc}(k, m):$	$\text{Dec}(k, c):$
$\mathcal{M} = \{0, 1\}^{\lambda+\ell}$	$k \leftarrow \mathcal{K}$	return $G(k) \oplus m$	return $G(k) \oplus c$
$\mathcal{C} = \{0, 1\}^{\lambda+\ell}$	return $k$		

### Claim:

If  $G$  is a secure PRG then pOTP[ $G$ ] satisfies one-time secrecy. That is,  $\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]} \equiv \mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$ .

# Overview:

Want to show:

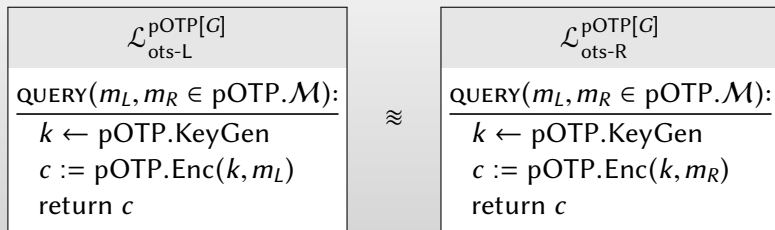


Standard hybrid technique:

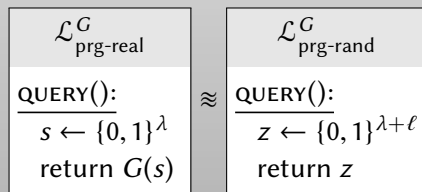
- ▶ Starting with  $\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]}$ , make a sequence of small modifications
- ▶ Each modification has *negligible* effect on calling program
- ▶ Sequence of modifications ends with  $\mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$

# Overview:

Want to show:



The proof will **use** the fact  $G$  is a secure PRG. In other words,



# Security proof


$$\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]}$$

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

---

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

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

return  $c$

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

# Security proof


$$\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]}$$

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

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

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

return  $c$

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

# Security proof



$\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]}$
$\text{QUERY}(m_L, m_R \in \{0, 1\}^{\lambda+\ell}):$
$k \leftarrow \{0, 1\}^\lambda$
$c := G(k) \oplus m_L$
return $c$

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

# Security proof


$$\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]}$$

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

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

$c := G(k) \oplus m_L$

return  $c$

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

# Security proof


$$\mathcal{L}_{\text{ots-L}}^{\text{pOTP}[G]}$$

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

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

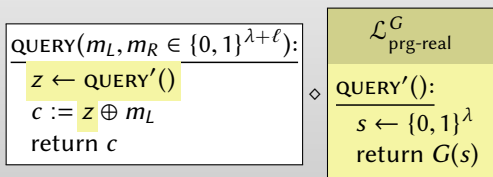
$c := G(k) \oplus m_L$

return  $c$

These statements appear in  $\mathcal{L}_{\text{prg-real}}^G$ .

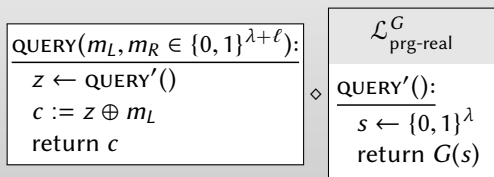


# Security proof



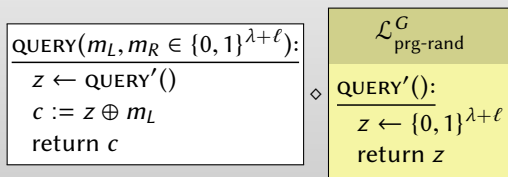
Factor out in terms of  $\mathcal{L}_{\text{prg-real}}^G$ .

# Security proof



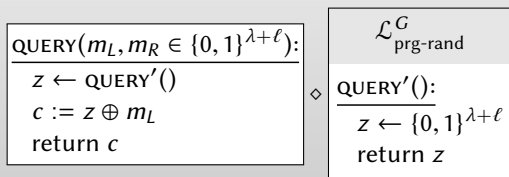
Factor out in terms of  $\mathcal{L}_{\text{prg-real}}^G$ .

# Security proof



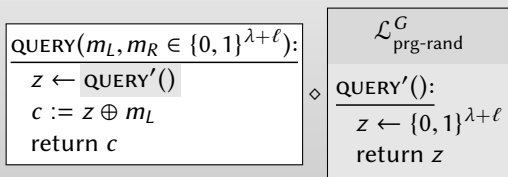
Security of  $G$  allows us to swap  $\mathcal{L}_{\text{prg-real}}^G$  with  $\mathcal{L}_{\text{prg-rand}}^G$ .

# Security proof



Security of  $G$  allows us to swap  $\mathcal{L}_{\text{prg-real}}^G$  with  $\mathcal{L}_{\text{prg-rand}}^G$ .

# Security proof



Inline call to  $\text{QUERY}'$ .

## Security proof



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

$z \leftarrow \{0, 1\}^{\lambda+\ell}$

$c := z \oplus m_L$

return  $c$

Inline call to QUERY'.

# Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^{\lambda+\ell}):$
$z \leftarrow \{0, 1\}^{\lambda+\ell}$
$c := z \oplus m_L$
return $c$

Inline call to QUERY'.

# Security proof



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

This is exactly  $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$  — one time pad!



# Security proof


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

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

---

 $k \leftarrow \{0, 1\}^{\lambda+\ell}$  $c := k \oplus m_L$ 

return  $c$

This is exactly  $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$  — one time pad!

# Security proof


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

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

---

 $k \leftarrow \{0, 1\}^{\lambda+\ell}$  $c := k \oplus m_R$ return  $c$ 

Security of one-time pad is that we can replace  $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$  with  $\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$ .

# Security proof


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

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

---

 $k \leftarrow \{0, 1\}^{\lambda+\ell}$  $c := k \oplus m_R$ 

return  $c$

Security of one-time pad is that we can replace  $\mathcal{L}_{\text{ots-L}}^{\text{OTP}}$  with  $\mathcal{L}_{\text{ots-R}}^{\text{OTP}}$ .

# Security proof


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

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

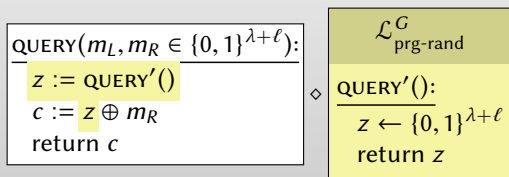
$k \leftarrow \{0, 1\}^{\lambda+\ell}$

$c := k \oplus m_R$

return  $c$

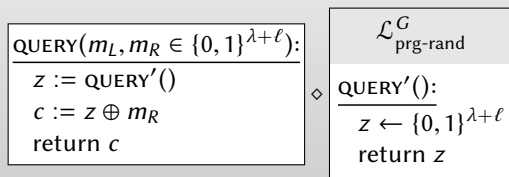
Same steps in reverse order. Factor out in terms of  $\mathcal{L}_{\text{prg-rand}}^G$ .

# Security proof



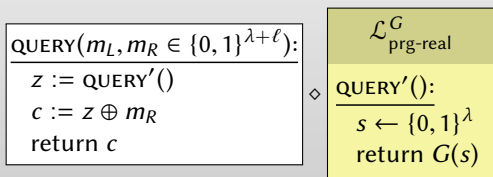
Same steps in reverse order. Factor out in terms of  $\mathcal{L}_{\text{prg-rand}}^G$ .

# Security proof



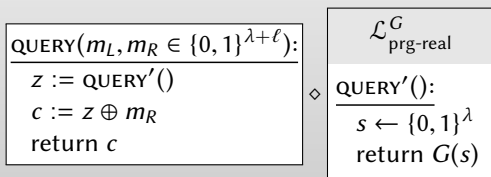
Same steps in reverse order. Factor out in terms of  $\mathcal{L}_{\text{prg-rand}}^G$ .

# Security proof



Replace  $\mathcal{L}_{\text{prg-rand}}^G$  with  $\mathcal{L}_{\text{prg-real}}^G$ .

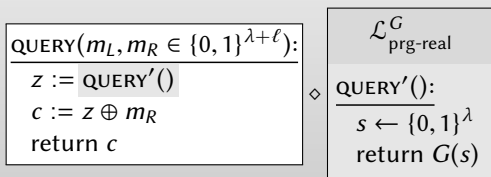
# Security proof



Replace  $\mathcal{L}_{\text{prg-rand}}^G$  with  $\mathcal{L}_{\text{prg-real}}^G$ .



# Security proof



Inline call to  $\text{QUERY}'$ .

# Security proof



```
QUERY( $m_L, m_R \in \{0, 1\}^{\lambda+\ell}$ ):  
-----  
 $k \leftarrow \{0, 1\}^\lambda$   
 $c := G(k) \oplus m_R$   
return  $c$ 
```

Inline call to QUERY'.

# Security proof



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

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

$c := G(k) \oplus m_R$

return  $c$

Inline call to QUERY'.

# Security proof



$\text{QUERY}(m_L, m_R \in \{0, 1\}^{\lambda+\ell}):$
$k \leftarrow \{0, 1\}^\lambda$
$c := G(k) \oplus m_R$
return $c$

This happens to be  $\mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$ .

# Security proof



$\mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$
QUERY( $m_L, m_R \in \text{pOTP}.\mathcal{K}$ ):
$k \leftarrow \text{pOTP}.\text{KeyGen}$
$c := \text{pOTP}.\text{Enc}(k, m_R)$
return $c$

This happens to be  $\mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$ .

# Security proof


$$\mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$$

QUERY( $m_L, m_R \in \text{pOTP}.\mathcal{K}$ ):

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

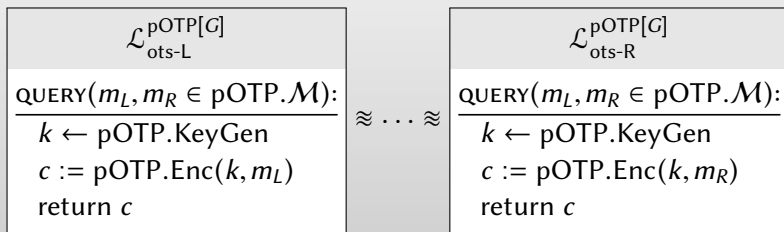
$c := \text{pOTP}.\text{Enc}(k, m_R)$

return  $c$

This happens to be  $\mathcal{L}_{\text{ots-R}}^{\text{pOTP}[G]}$ .

# Summary

We showed:



So  $\text{pOTP}[G]$  satisfies one-time secrecy when  $G$  is a secure PRG.