How Secure is Deterministic Encryption?

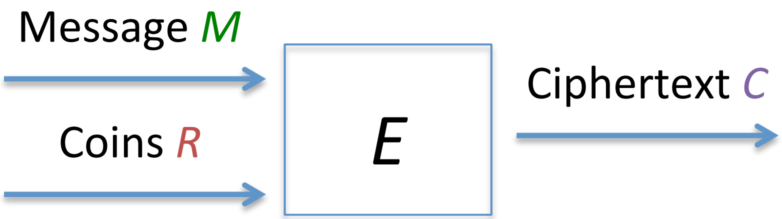
Mihir Bellare
Rafael Dowsley
Sriram Keelveedhi

Deterministic Encryption

Deterministic encryption was introduced by Bellare, Boldyreva and O'neill [BBO07] and offers practical benefits in certain applications such as efficient search on encrypted databases [BBO07] and resilience in the face of low-quality randomness that occurs in many systems [BBN+09,RY10].

Deterministic Encryption

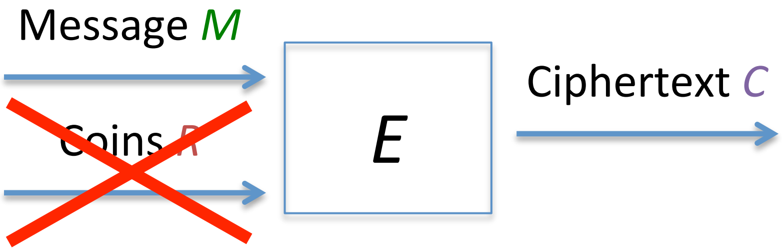
Deterministic encryption was introduced by Bellare, Boldyreva and O'neill [BBO07] and offers practical benefits in certain applications such as efficient search on encrypted databases [BBO07] and resilience in the face of low-quality randomness that occurs in many systems [BBN+09,RY10].



Security Goal: IND-CPA, IND-CCA2

Deterministic Encryption

Deterministic encryption was introduced by Bellare, Boldyreva and O'neill [BBO07] and offers practical benefits in certain applications such as efficient search on encrypted databases [BBO07] and resilience in the face of low-quality randomness that occurs in many systems [BBN+09,RY10].



Message M

~~Coins R~~

E

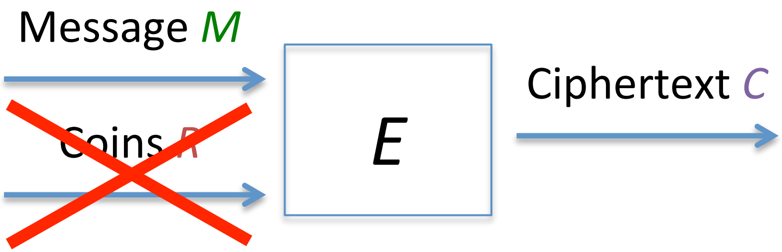
Ciphertext C

Security Goal: ~~IND-CPA, IND-CCA2~~

Best Possible!

Deterministic Encryption

Deterministic encryption was introduced by Bellare, Boldyreva and O’neill [BBO07] and offers practical benefits in certain applications such as efficient search on encrypted databases [BBO07] and resilience in the face of low-quality randomness that occurs in many systems [BBN+09,RY10].



Message M

~~Coins R~~

E

Ciphertext C

Security Goal: ~~IND-CPA, IND-CCA2~~

Best Possible!

Security can be formalized using the PRIV definition [BBO07] or equivalently an IND-style definition [BFOR08], but these definitions are unusual.

Security Definition

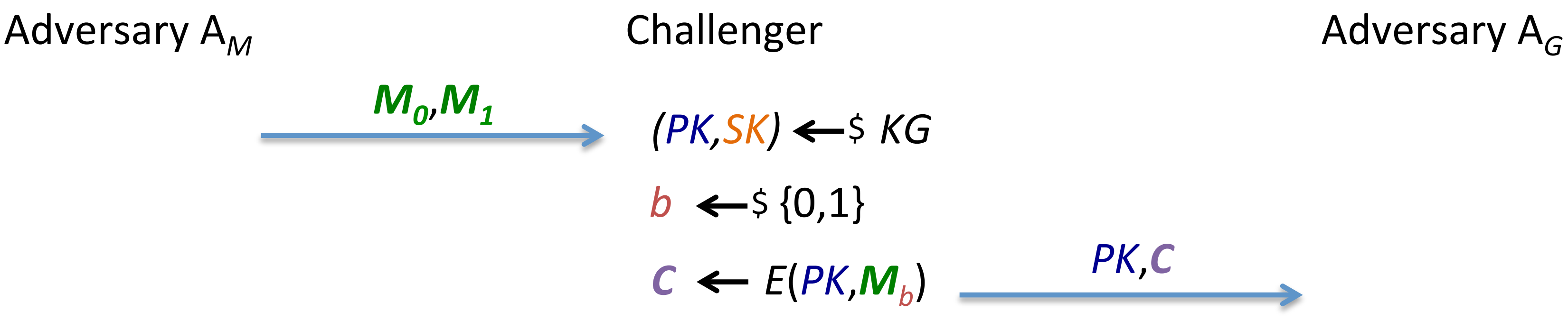
Adversary A_M

Challenger

Adversary A_G

M_0,M_1

Security Definition



Adversary A_M

Challenger

Adversary A_G

M_0,M_1

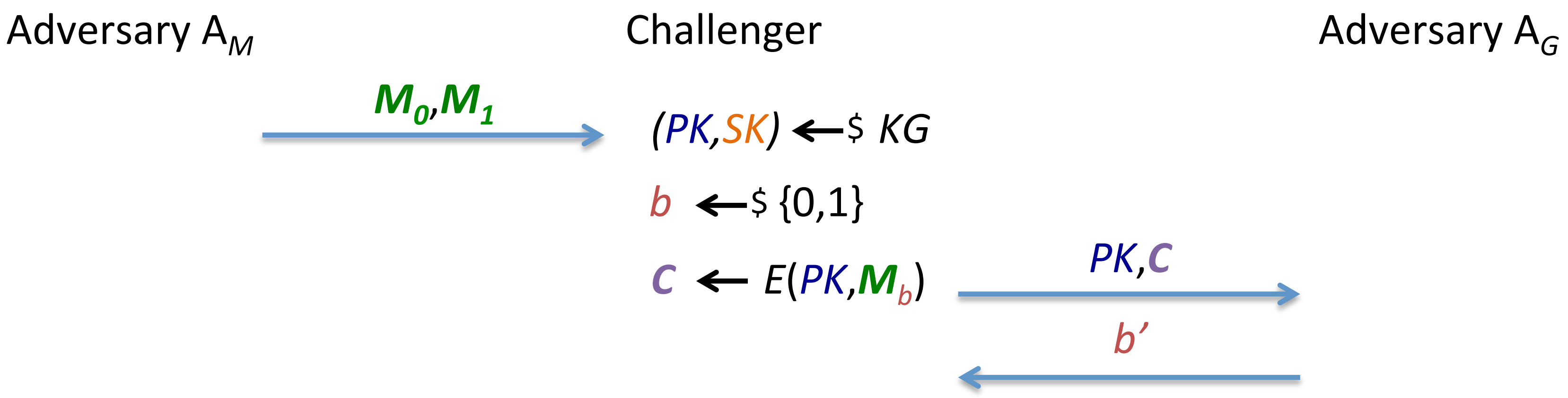
$(PK,SK) \leftarrow\$\ KG$

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

$C \leftarrow E(PK,M_b)$

PK,C

Security Definition



Adversary A_M

Challenger

Adversary A_G

M_0,M_1

$(PK,SK) \leftarrow\$ \; KG$

$b \leftarrow\$ \; \{0,1\}$

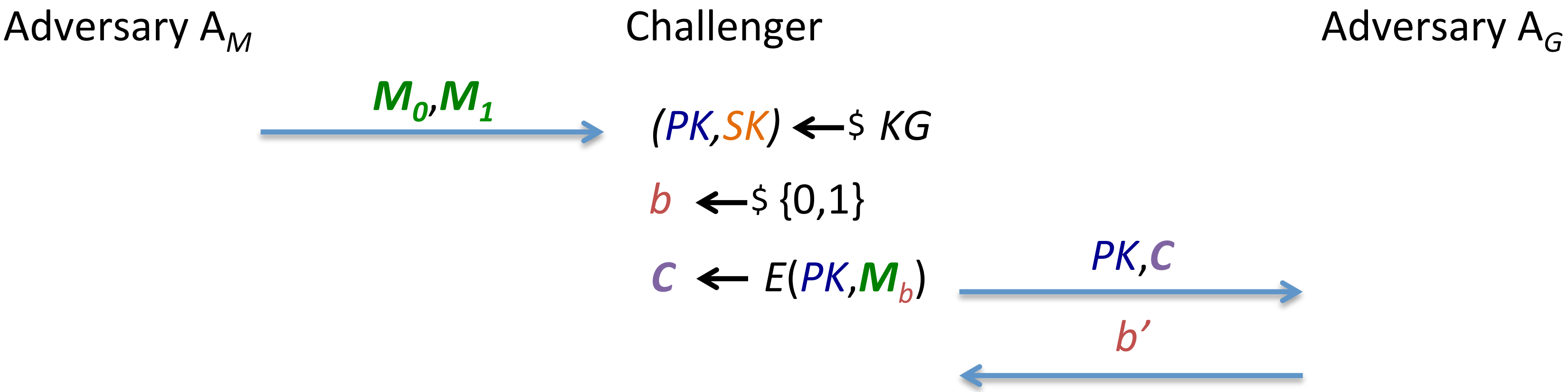
$C \leftarrow E(PK,M_b)$

PK,C

b'

The adversary (A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

Security Definition



Adversary A_M

Challenger

Adversary A_G

M_0,M_1

$(PK,SK) \leftarrow\$ KG$

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

$C \leftarrow E(PK,M_b)$

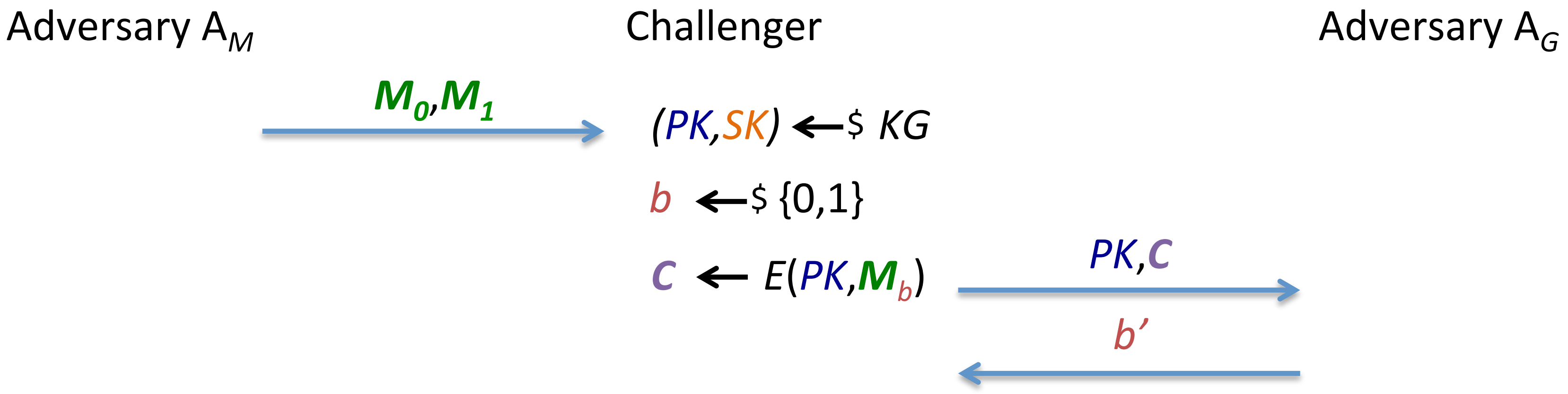
PK,C

b'

The adversary (A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

There are three essential restrictions without which security is not achievable:

Security Definition



Adversary A_M

Challenger

Adversary A_G

M_0,M_1

$(PK,SK) \leftarrow\$\ KG$

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

$C \leftarrow E(PK,M_b)$

PK,C

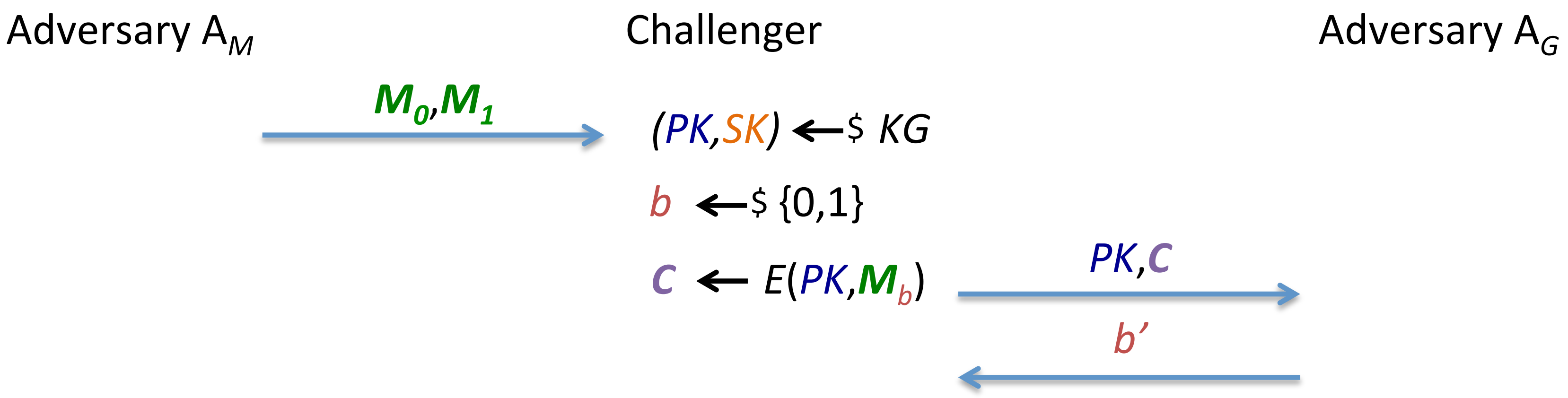
b'

The adversary (A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

There are three essential restrictions without which security is not achievable:

1) A_M does not get the public key.

Security Definition



Adversary A_M

Challenger

Adversary A_G

M_0,M_1

$(PK,SK) \leftarrow\$ KG$

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

$C \leftarrow E(PK,M_b)$

PK,C

b'

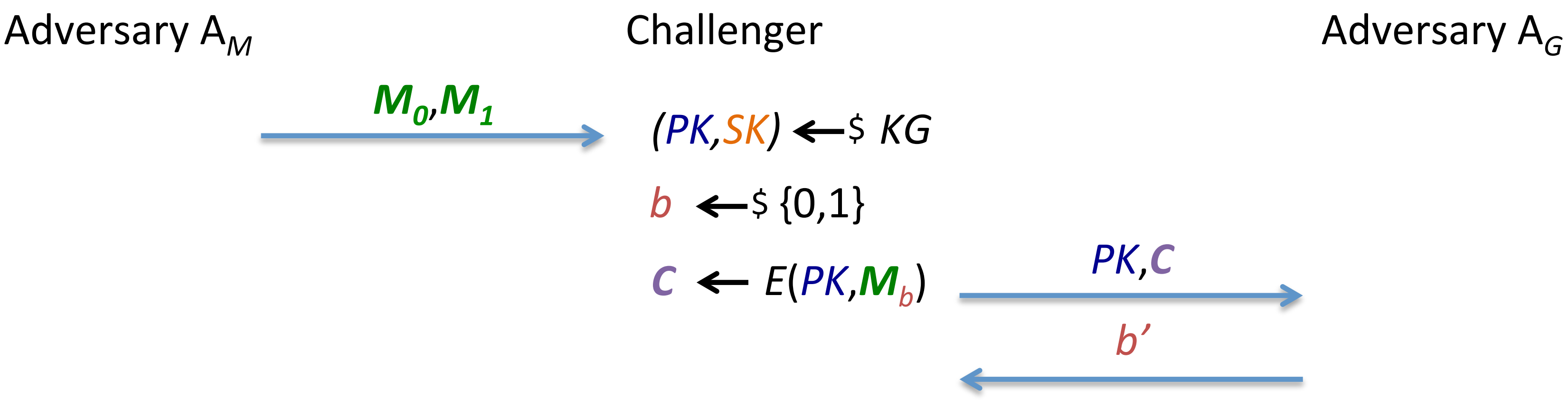
The adversary (A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

There are three essential restrictions without which security is not achievable:

1) A_M does not get the public key.

2) All messages in M_0,M_1 must have high min-entropy and there are no repeated messages in the same vector.

Security Definition



Adversary A_M

Challenger

Adversary A_G

M_0,M_1

$(PK,SK) \leftarrow\$\ KG$

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

$C \leftarrow E(PK,M_b)$

PK,C

b'

The adversary (A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

There are three essential restrictions without which security is not achievable:

1) A_M does not get the public key.

2) All messages in M_0,M_1 must have high min-entropy and there are no repeated messages in the same vector.

3) A_M cannot pass state to A_G.

Questions

Does security in the standard model implies security in the random oracle model?

Questions

Does security in the standard model implies security in the random oracle model?

Is it possible to achieve security against selective opening attacks?

Questions

Does security in the standard model implies security in the random oracle model?

Is it possible to achieve security against selective opening attacks?

Does single-user security implies multi-user security?

Questions

Does security in the standard model implies security in the random oracle model?

Is it possible to achieve security against selective opening attacks?

Does single-user security implies multi-user security?

In the case of randomized PKE the answer to these questions is YES, but for deterministic encryption the situation is different and our results will show some subtle points about security definitions for deterministic PKE.

Does security in the standard model implies security in the random oracle model?

Tautology?

Intuitively, it seems clear that security in the standard model should imply security in the random oracle (RO) model. If the scheme does not use the RO, then given the adversary access to the RO cannot violate security.

Tautology?

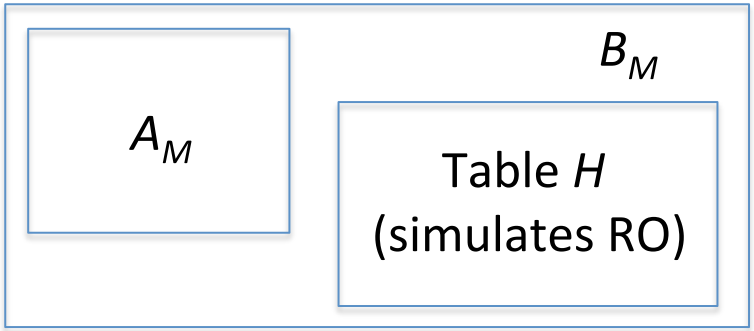
Intuitively, it seems clear that security in the standard model should imply security in the random oracle (RO) model. If the scheme does not use the RO, then given the adversary access to the RO cannot violate security.

For randomized PKE this is true and can be formalized. Given an random oracle adversary (A_M,A_G) it is possible to build a standard model adversary (B_M,B_G) with the same advantage.

Tautology?

Intuitively, it seems clear that security in the standard model should imply security in the random oracle (RO) model. If the scheme does not use the RO, then given the adversary access to the RO cannot violate security.

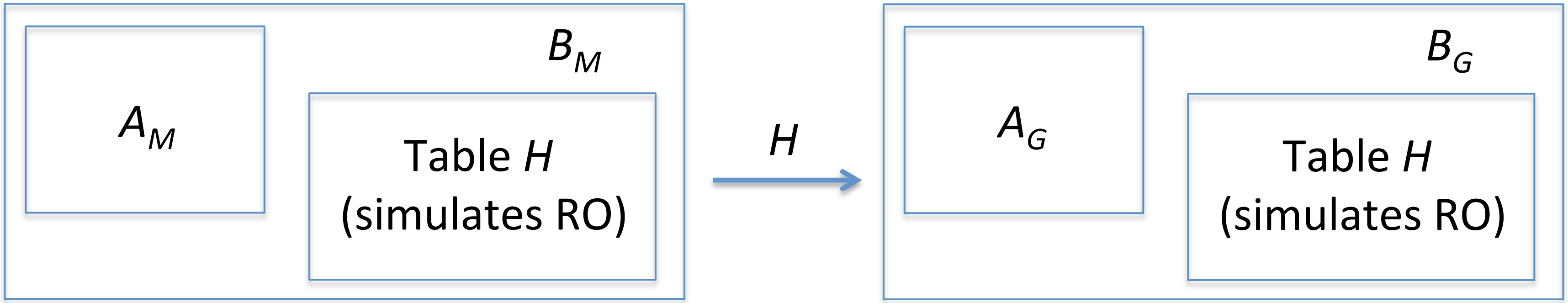
For randomized PKE this is true and can be formalized. Given an random oracle adversary (A_M,A_G) it is possible to build a standard model adversary (B_M,B_G) with the same advantage.



Tautology?

Intuitively, it seems clear that security in the standard model should imply security in the random oracle (RO) model. If the scheme does not use the RO, then given the adversary access to the RO cannot violate security.

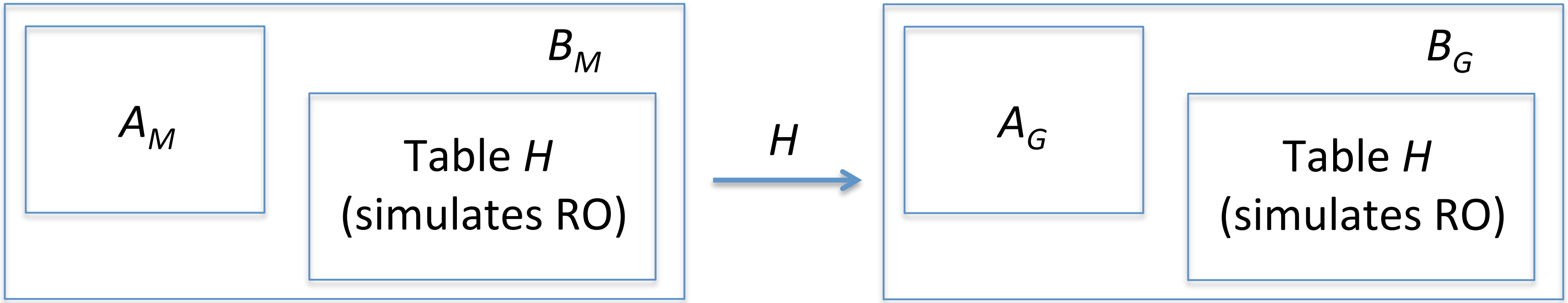
For randomized PKE this is true and can be formalized. Given an random oracle adversary (A_M,A_G) it is possible to build a standard model adversary (B_M,B_G) with the same advantage.



Tautology?

Intuitively, it seems clear that security in the standard model should imply security in the random oracle (RO) model. If the scheme does not use the RO, then given the adversary access to the RO cannot violate security.

For randomized PKE this is true and can be formalized. Given an random oracle adversary (A_M,A_G) it is possible to build a standard model adversary (B_M,B_G) with the same advantage.



B_M

A_M

Table H
(simulates RO)

H

B_G

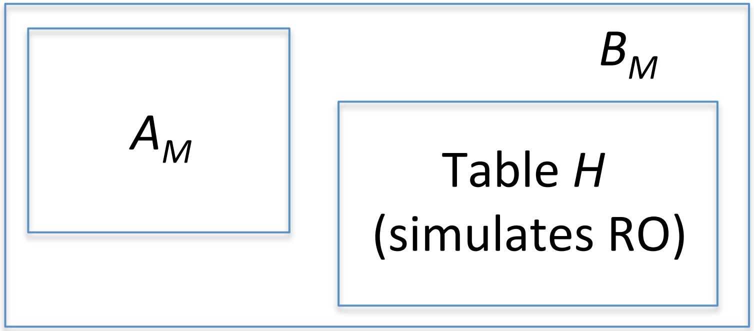
A_G

Table H
(simulates RO)

The claim and the simulation argument hardly seem specific to randomized PKE.

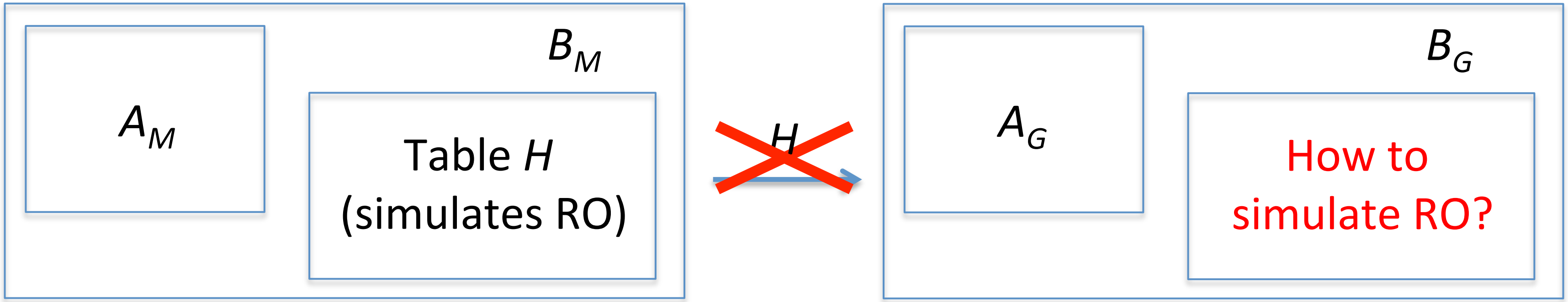
The Case of Deterministic PKE

Lets consider a deterministic PKE. Given an random oracle adversary (A_M,A_G) we try to build a standard model adversary (B_M,B_G) using the previous technique.



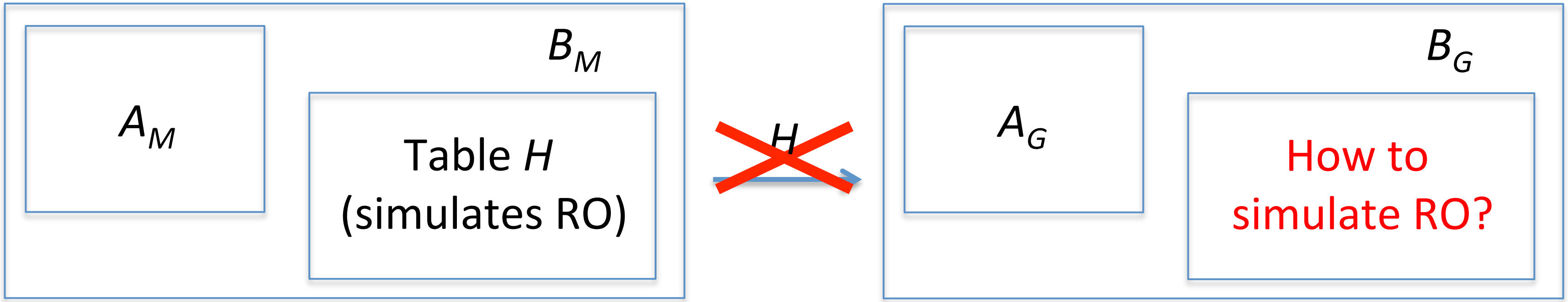
The Case of Deterministic PKE

Lets consider a deterministic PKE. Given an random oracle adversary (A_M, A_G) we try to build a standard model adversary (B_M, B_G) using the previous technique.



The Case of Deterministic PKE

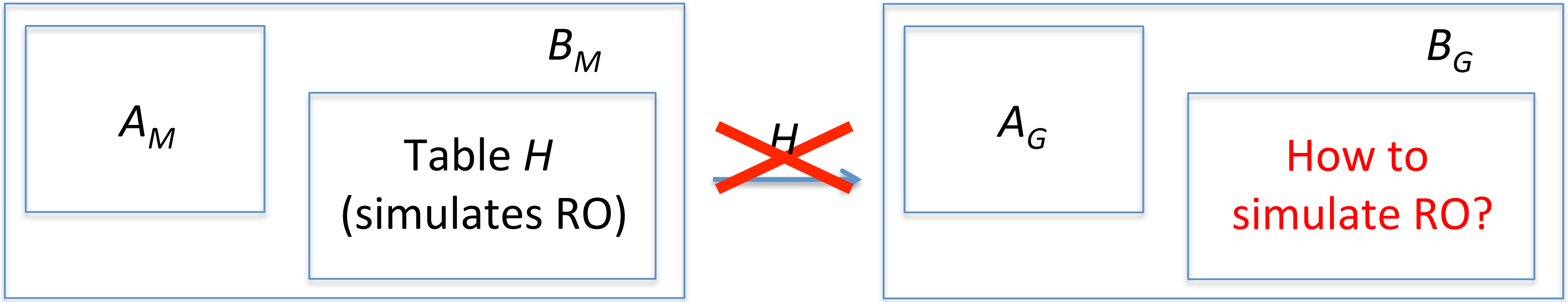
Lets consider a deterministic PKE. Given an random oracle adversary (A_M, A_G) we try to build a standard model adversary (B_M, B_G) using the previous technique.



If B_G simulates a new RO for A_G, then this is not coherent with what (A_M, A_G) gets in the real random oracle model game.

The Case of Deterministic PKE

Lets consider a deterministic PKE. Given an random oracle adversary (A_M, A_G) we try to build a standard model adversary (B_M, B_G) using the previous technique.



If B_G simulates a new RO for A_G, then this is not coherent with what (A_M, A_G) gets in the real random oracle model game.

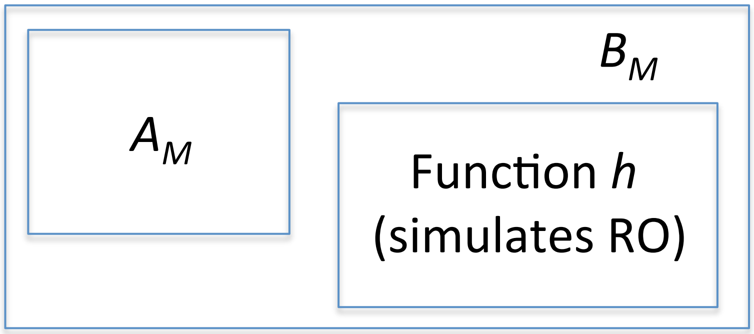
It is not completely clear if the security implication always holds for deterministic PKE: whether we could prove it or not depended on details of the security definition.

Non-Uniform Adversaries

Idea: use a q-wise independent hash function h to simulate the random oracle. Hardwire h into the circuits of B_m and B_G.

Non-Uniform Adversaries

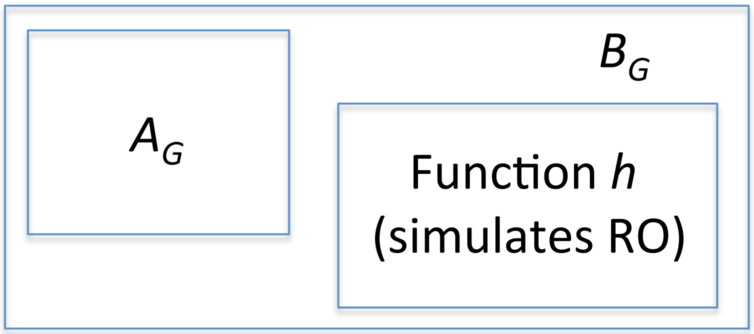
Idea: use a q-wise independent hash function h to simulate the random oracle. Hardwire h into the circuits of B_m and B_G.



B_M

A_M

Function h
(simulates RO)



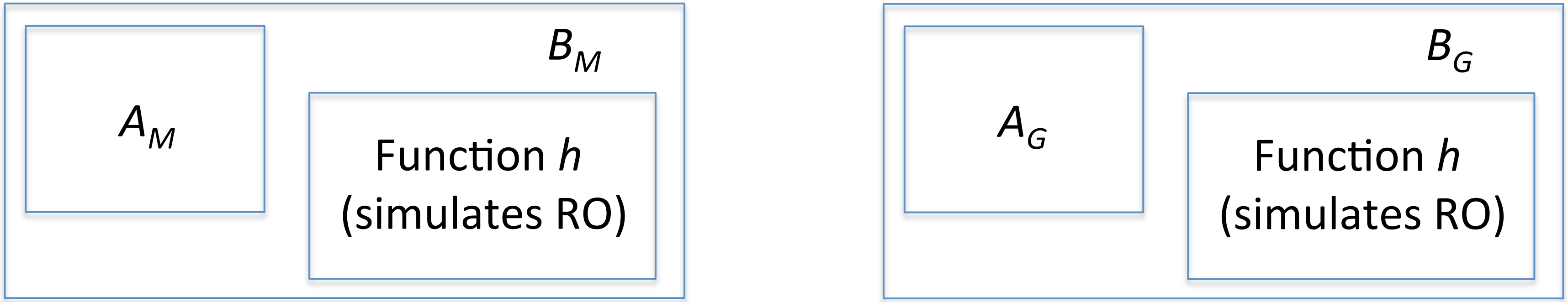
B_G

A_G

Function h
(simulates RO)

Non-Uniform Adversaries

Idea: use a q-wise independent hash function h to simulate the random oracle. Hardwire h into the circuits of B_m and B_G.



B_M

A_M

Function h (simulates RO)

B_G

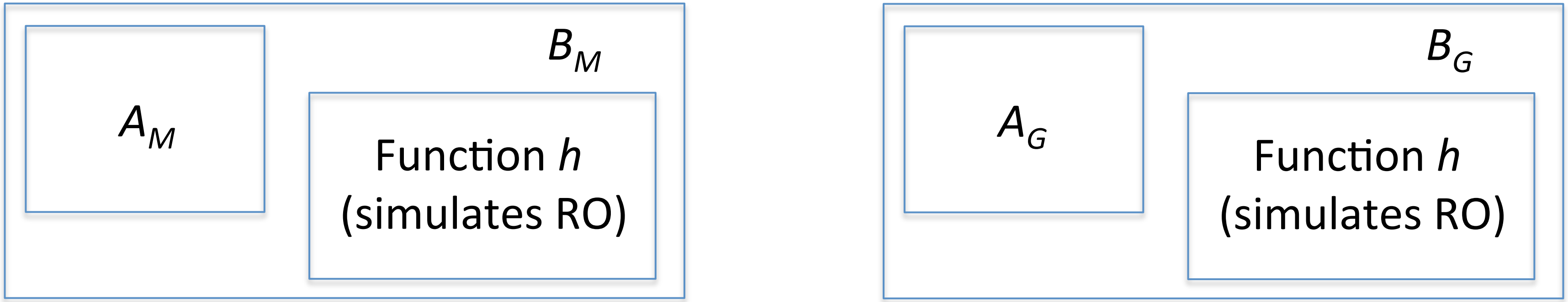
A_G

Function h (simulates RO)

Possible to show that standard model security implies RO model security.

Non-Uniform Adversaries

Idea: use a q-wise independent hash function h to simulate the random oracle. Hardwire h into the circuits of B_m and B_G.



B_M

A_M

Function h (simulates RO)

B_G

A_G

Function h (simulates RO)

Possible to show that standard model security implies RO model security.

Note that q-wise independence is a non-adaptive condition, while the RO queries are adaptive. But it is possible to handle this in the analysis.

Uniform Adversaries

For uniform adversaries it is not clear how to perform a simulation/do a proof.

Uniform Adversaries

For uniform adversaries it is not clear how to perform a simulation/do a proof.

But we also cannot imagine a counter-example.

Uniform Adversaries

For uniform adversaries it is not clear how to perform a simulation/do a proof.

But we also cannot imagine a counter-example.

A counter-example would need to exploit the fact that a scheme is secure against uniform adversaries, but not against non-uniform ones.

Uniform Adversaries

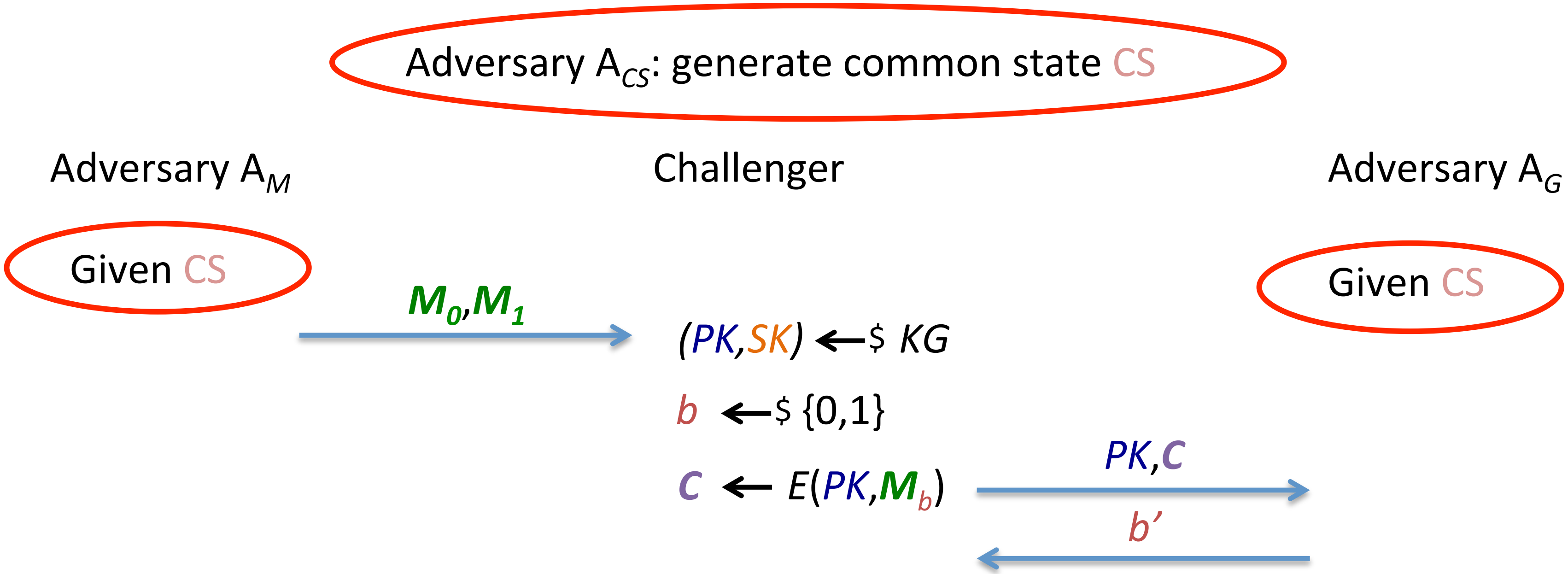
For uniform adversaries it is not clear how to perform a simulation/do a proof.

But we also cannot imagine a counter-example.

A counter-example would need to exploit the fact that a scheme is secure against uniform adversaries, but not against non-uniform ones.

Intuitively it is hard to imagine how a standard model scheme can be insecure in the RO model if the messages have high min-entropy conditioned on the RO.

Three Stage Adversaries



Adversary A_{CS}: generate common state CS

Adversary A_M

Challenger

Adversary A_G

Given CS

M_0,M_1

$(PK,SK) \leftarrow\$ KG$

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

$C \leftarrow E(PK,M_b)$

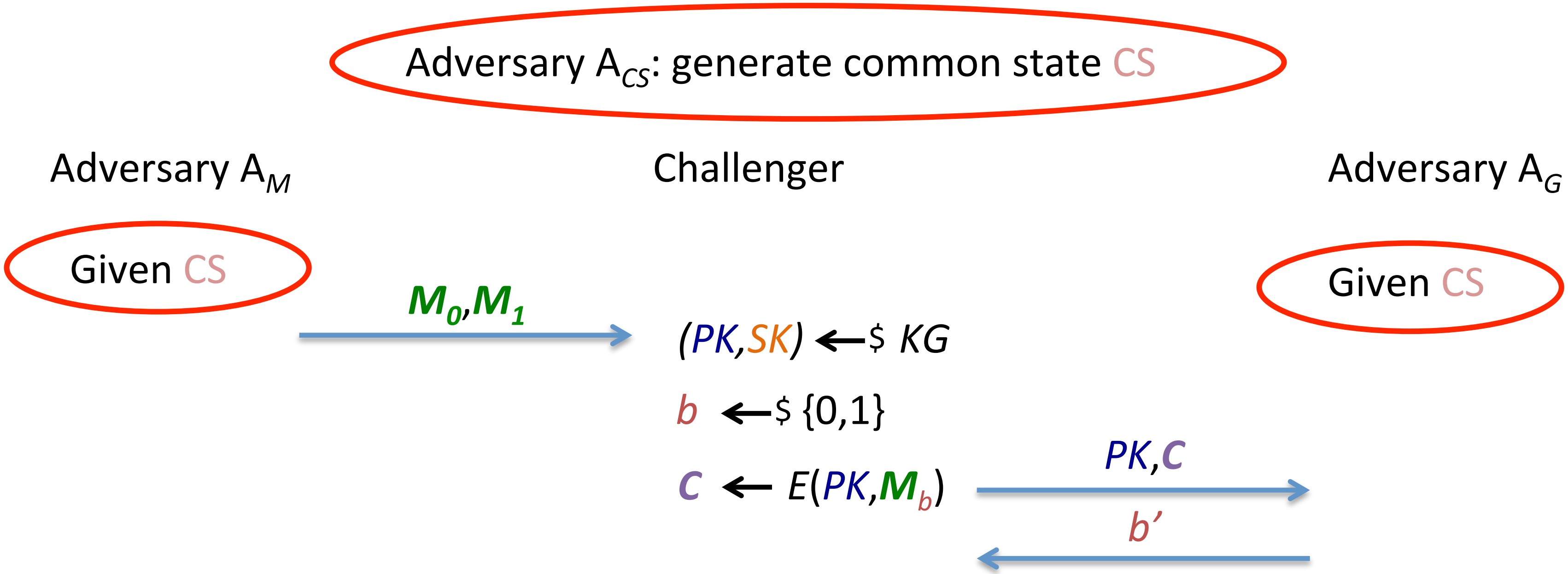
PK,C

b'

Given CS

Adversary (A_{CS},A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

Three Stage Adversaries



Adversary A_{CS}: generate common state CS

Adversary A_M

Given CS

M_0,M_1

Challenger

$(PK,SK) \leftarrow\$ KG$

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

$C \leftarrow E(PK,M_b)$

PK,C

b'

Adversary A_G

Given CS

Adversary (A_{CS},A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

There are three essential restrictions without which security is not achievable:

1) A_{CS} and A_M do not get the public key.

2) All messages in M_0,M_1 must have high min-entropy and there are no repeated messages in the same vector.

3) A_M cannot pass state to A_G.

Three Stage Adversaries

If the definition with three stage adversaries is considered, then security in the standard model implies security in the random oracle model.

Three Stage Adversaries

If the definition with three stage adversaries is considered, then security in the standard model implies security in the random oracle model.

The idea is that the common state can include the key for a q-wise independent family of functions.

Three Stage Adversaries

If the definition with three stage adversaries is considered, then security in the standard model implies security in the random oracle model.

The idea is that the common state can include the key for a q-wise independent family of functions.

Takeaway: Use the definition with three stage adversaries.

Is it possible to achieve security against selective opening attacks?

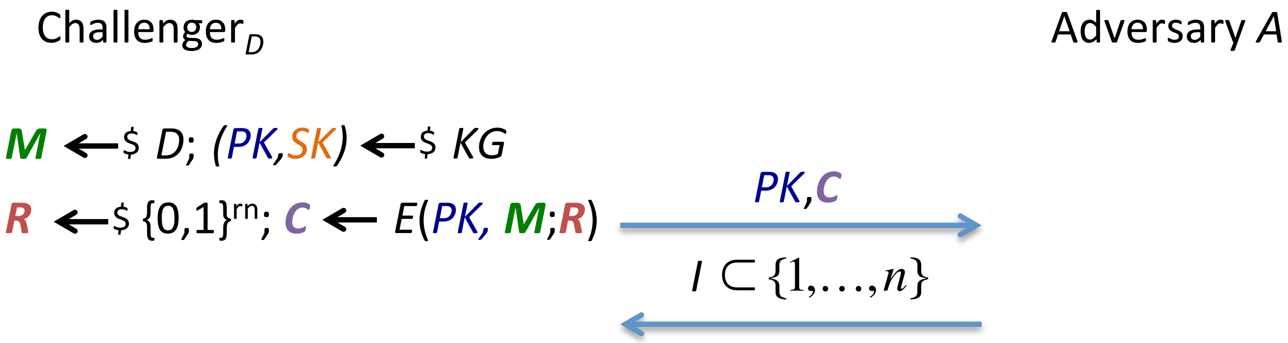
SOA-M

Challenger$_D$ Adversary A

$M \leftarrow\$ \; D$; $(PK,SK) \leftarrow\$ \; KG$

$R \leftarrow\$ \; \{0,1\}^{rn}$; $C \leftarrow E(PK, M;R)$ $\xrightarrow{PK,C}$

SOA-M



Challenger_D

Adversary A

$M \leftarrow_{\$} D$; $(PK, SK) \leftarrow_{\$} KG$

$R \leftarrow_{\$} \{0,1\}^{rn}$; $C \leftarrow E(PK, M; R)$

$$PK, C \longrightarrow$$

$$I \subset \{1,\dots,n\} \longleftarrow$$

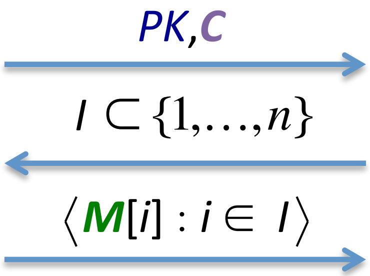
SOA-M

Challenger$_D$

Adversary A

$M \leftarrow\$ D$; $(PK,SK) \leftarrow\$ KG$

$R \leftarrow\$ \{0,1\}^{rn}$; $C \leftarrow E(PK, M;R)$

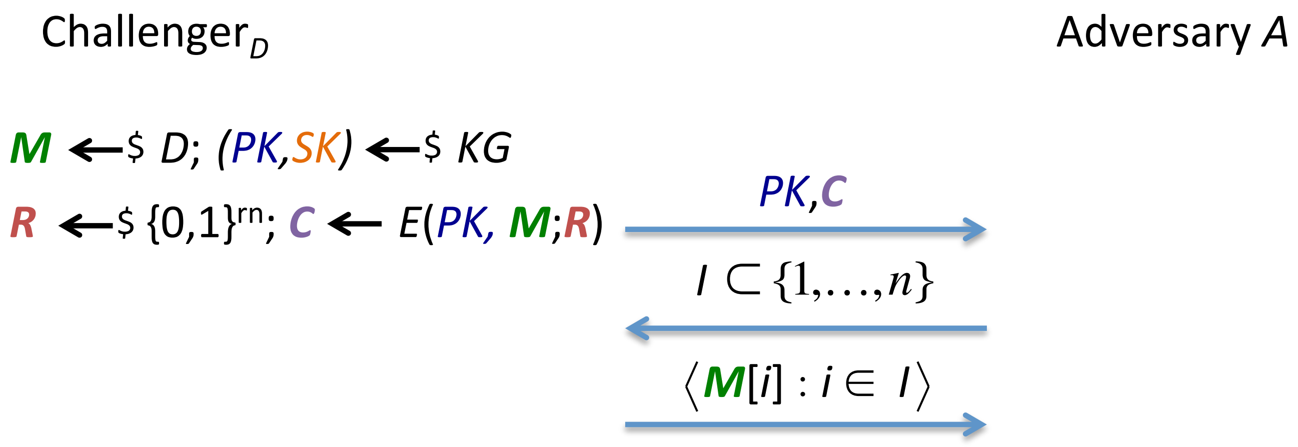


PK,C

$I \subset \{1,\ldots,n\}$

$\langle M[i] : i \in I \rangle$

SOA-M



Challenger$_D$

Adversary A

$M \leftarrow\$ D$; $(PK,SK) \leftarrow\$ KG$

$R \leftarrow\$ \{0,1\}^{rn}$; $C \leftarrow E(PK, M;R)$

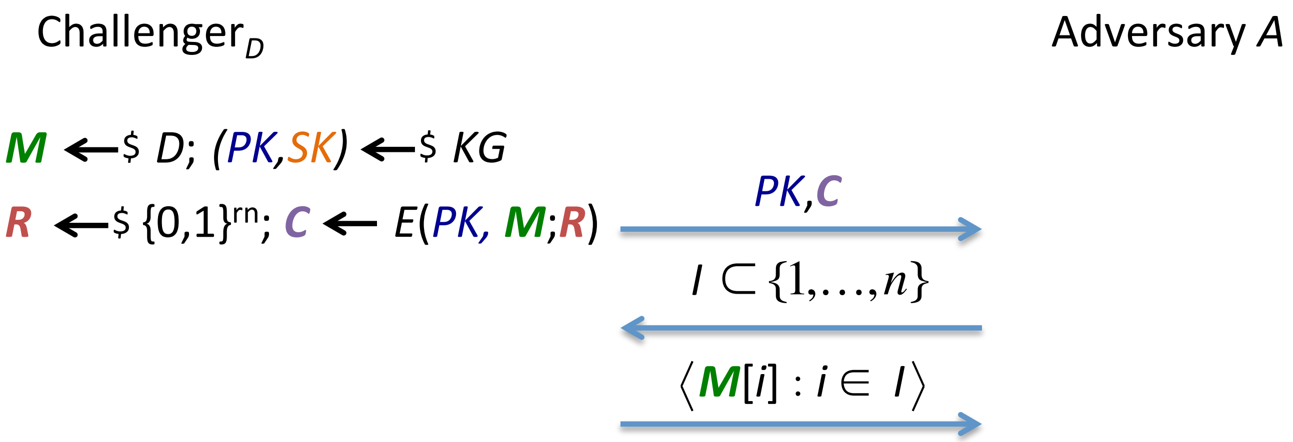
$\xrightarrow{PK,C}$

$\xleftarrow{I \subset \{1,\ldots,n\}}$

$\xrightarrow{\langle M[i] : i \in I \rangle}$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

SOA-M



Challenger$_D$

Adversary A

$M \xleftarrow{\$} D$; $(PK,SK) \xleftarrow{\$} KG$

$R \xleftarrow{\$} \{0,1\}^{rn}$; $C \leftarrow E(PK, M;R)$

$\xrightarrow{PK,C}$

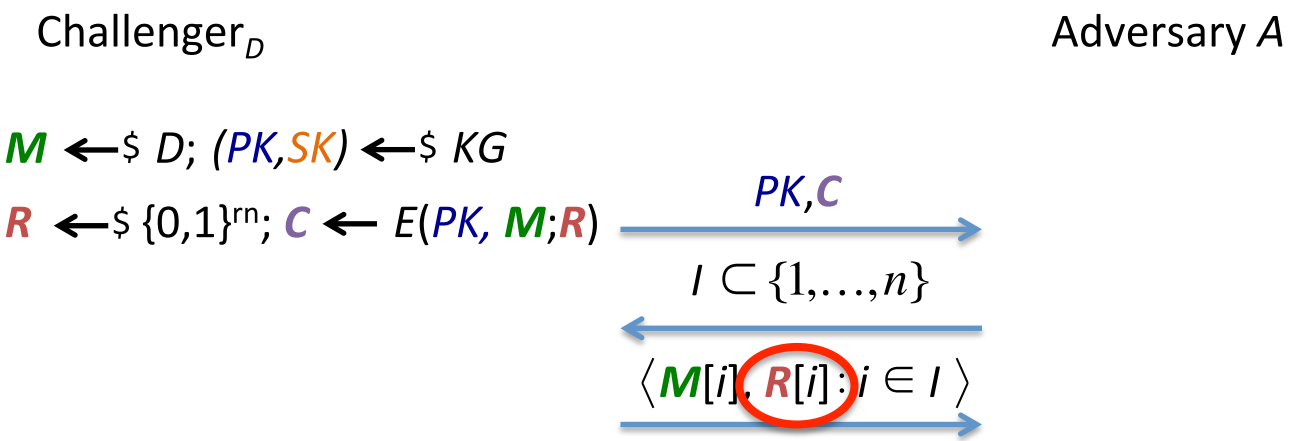
$\xleftarrow{I \subset \{1,\ldots,n\}}$

$\xrightarrow{\langle M[i] \,:\, i \in I \rangle}$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

IND-CPA security implies SOA-M security [BY09].

SOA-C



Challenger$_D$

Adversary A

$M \xleftarrow{\$} D$; $(PK,SK) \xleftarrow{\$} KG$

$R \xleftarrow{\$} \{0,1\}^{rn}$; $C \leftarrow E(PK, M;R)$

$\xrightarrow{PK,C}$

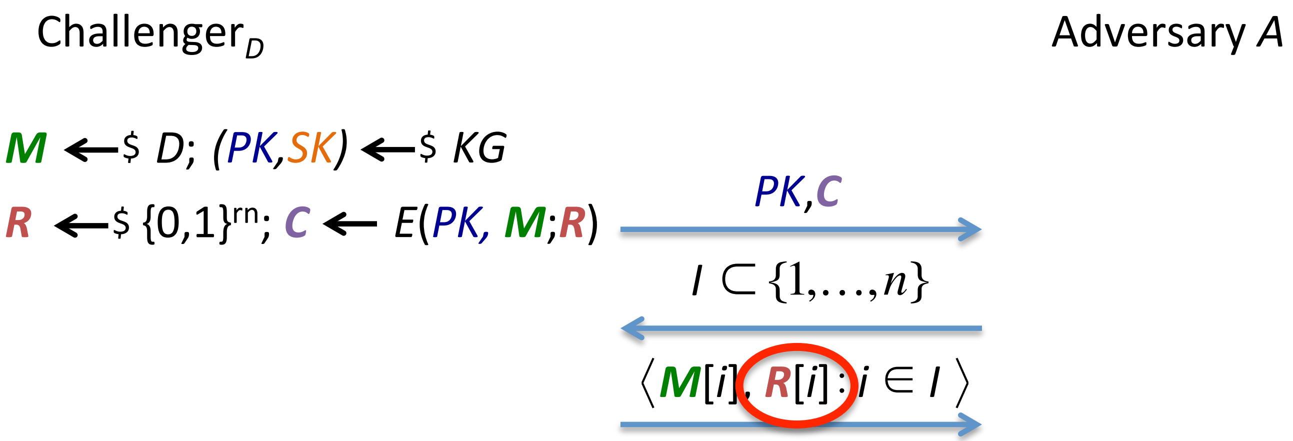
$\xleftarrow{I \subset \{1,\ldots,n\}}$

$\xrightarrow{\langle M[i], \bigcirc\!\!\!\!R[i] : i \in I \rangle}$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

IND-CPA security implies SOA-M security [BY09].

SOA-C



Challenger$_D$

Adversary A

$M \xleftarrow{\$} D$; $(PK, SK) \xleftarrow{\$} KG$

$R \xleftarrow{\$} \{0,1\}^{rn}$; $C \leftarrow E(PK, M; R)$

$\xrightarrow{PK, C}$

$\xleftarrow{I \subset \{1,\ldots,n\}}$

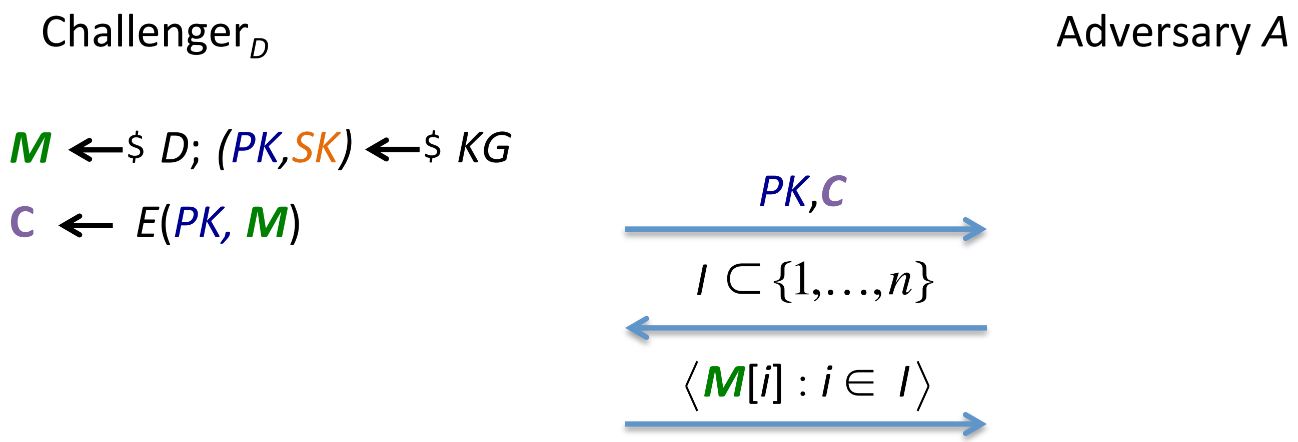
$\xrightarrow{\langle M[i], R[i] : i \in I \rangle}$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

IND-CPA security implies SOA-M security [BY09].

While IND-CPA (or even IND-CCA2) does not imply SOA-C security [BDWY12], it is possible to achieve SOA-C security [BHY09].

SOA for Deterministic PKE



Challenger$_D$

Adversary A

$M \xleftarrow{\$} D$; $(PK,SK) \xleftarrow{\$} KG$

$C \leftarrow E(PK, M)$

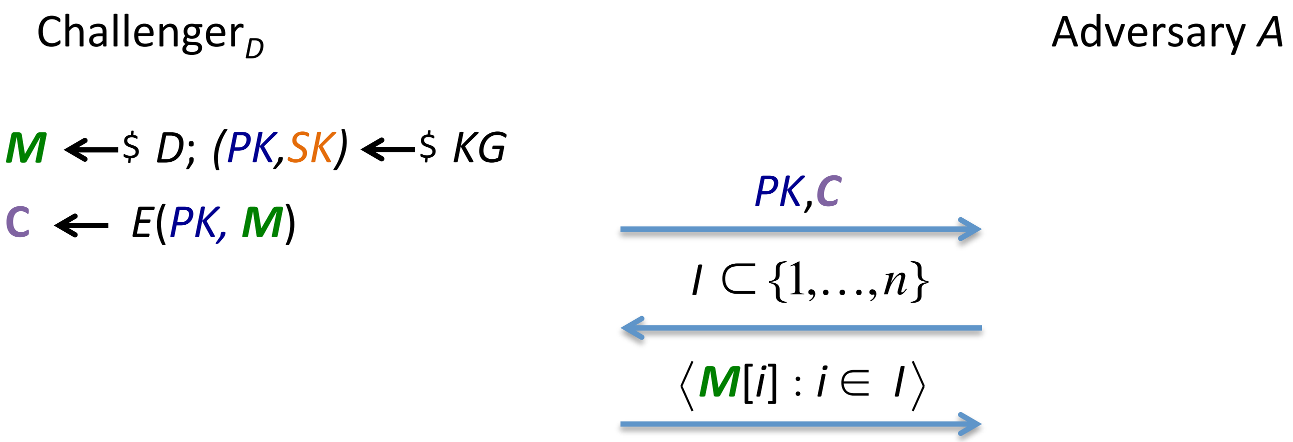
PK, C

$I \subset \{1,\ldots,n\}$

$\langle M[i] : i \in I \rangle$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

SOA for Deterministic PKE



Challenger$_D$

Adversary A

$M \leftarrow\$ D$; $(PK,SK) \leftarrow\$ KG$

$C \leftarrow E(PK, M)$

PK, C

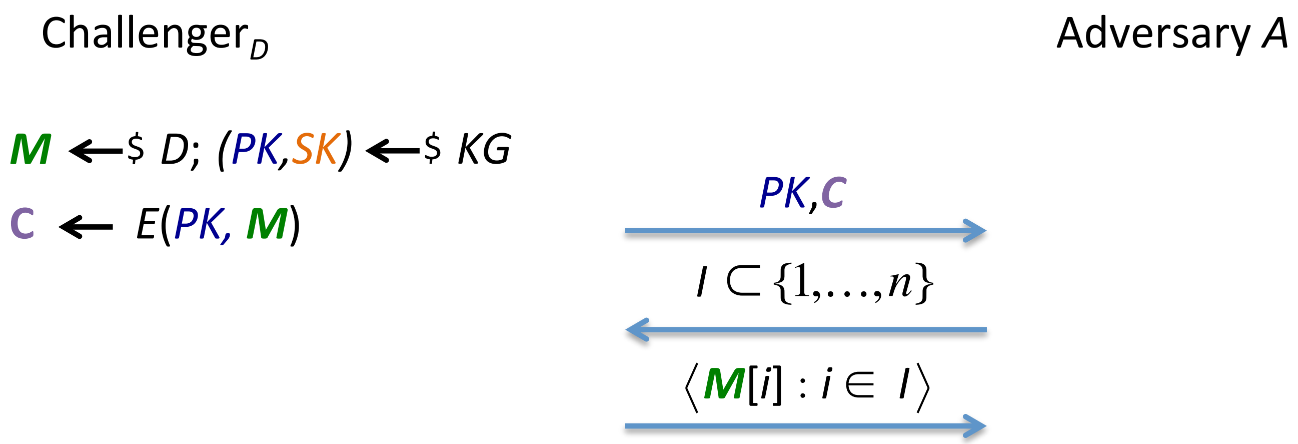
$I \subset \{1,\ldots,n\}$

$\langle M[i] : i \in I \rangle$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

Since the difficulty of achieving SOA security for randomized PKE lies in exposure of the coins, one might get the impression that SOA-security would be trivial to achieve for deterministic PKEs.

SOA for Deterministic PKE



Challenger$_D$

Adversary A

$M \leftarrow\$ \ D$; $(PK, SK) \leftarrow\$ \ KG$

$C \leftarrow E(PK, M)$

$$\xrightarrow{PK, C}$$

$$\xleftarrow{I \subset \{1,\ldots,n\}}$$

$$\xrightarrow{\langle M[i] : i \in I \rangle}$$

Security means A cannot figure out anything about $\langle M[i] : i \notin I \rangle$.

Since the difficulty of achieving SOA security for randomized PKE lies in exposure of the coins, one might get the impression that SOA-security would be trivial to achieve for deterministic PKEs.

Contrary is true: unachievable.

SOA Security Definition

Formalized using a simulation-based definition.

SOA Security Definition

Formalized using a simulation-based definition.

Uses the weaker semantic security for functions, instead of semantic security for relations that is used for randomized PKE.

SOA Security Definition

Formalized using a simulation-based definition.

Uses the weaker semantic security for functions, instead of semantic security for relations that is used for randomized PKE.

Same restrictions on the messages as before.

SOA Security Definition

Formalized using a simulation-based definition.

Uses the weaker semantic security for functions, instead of semantic security for relations that is used for randomized PKE.

Same restrictions on the messages as before.

The adversary in our result actually uses uniform, independent messages.

Result

In the case of randomized PKE, Bellare, Dowsley, Waters and Yilek [BDWY12] showed that any scheme satisfying a certain binding property is not SOA-secure.

Result

In the case of randomized PKE, Bellare, Dowsley, Waters and Yilek [BDWY12] showed that any scheme satisfying a certain binding property is not SOA-secure.

That binding property roughly requires the scheme to remain injective even on dishonestly-chosen public keys.

Result

In the case of randomized PKE, Bellare, Dowsley, Waters and Yilek [BDWY12] showed that any scheme satisfying a certain binding property is not SOA-secure.

That binding property roughly requires the scheme to remain injective even on dishonestly-chosen public keys.

For deterministic PKE, we show that every scheme admits a verification algorithm that tests the extent to which the encryption induced by a public key (even dishonestly-chosen ones) is an injective function. If it is far from injective, it gets detected, otherwise we have some sort of binding.

Result

In the case of randomized PKE, Bellare, Dowsley, Waters and Yilek [BDWY12] showed that any scheme satisfying a certain binding property is not SOA-secure.

That binding property roughly requires the scheme to remain injective even on dishonestly-chosen public keys.

For deterministic PKE, we show that every scheme admits a verification algorithm that tests the extent to which the encryption induced by a public key (even dishonestly-chosen ones) is an injective function. If it is far from injective, it gets detected, otherwise we have some sort of binding.

Adapt technique of Bellare et al. to show that no deterministic PKE is SOA-secure.

IND-Style Definition

A natural question is whether SOA-security for deterministic PKE can be achieved under a weaker, IND-style definition.

IND-Style Definition

A natural question is whether SOA-security for deterministic PKE can be achieved under a weaker, IND-style definition.

Not clear how to give a meaningful IND-style definition.

IND-Style Definition

A natural question is whether SOA-security for deterministic PKE can be achieved under a weaker, IND-style definition.

Not clear how to give a meaningful IND-style definition.

For randomized PKE, the IND-style definition involves conditional re-sampling of the un-opened messages. But for deterministic PKE we cannot provide the un-opened messages in the distinguishing test since the adversary could easily win by re-encrypting to check versus the ciphertexts.

IND-Style Definition

A natural question is whether SOA-security for deterministic PKE can be achieved under a weaker, IND-style definition.

Not clear how to give a meaningful IND-style definition.

For randomized PKE, the IND-style definition involves conditional re-sampling of the un-opened messages. But for deterministic PKE we cannot provide the un-opened messages in the distinguishing test since the adversary could easily win by re-encrypting to check versus the ciphertexts.

Problems even for randomized PKE: very limited set of message distributions or non-polynomial time games.

Does single-user security implies multi-user security?

mIND Security

Adversary A_{CS}: generate common state CS

Adversary A_M

Challenger

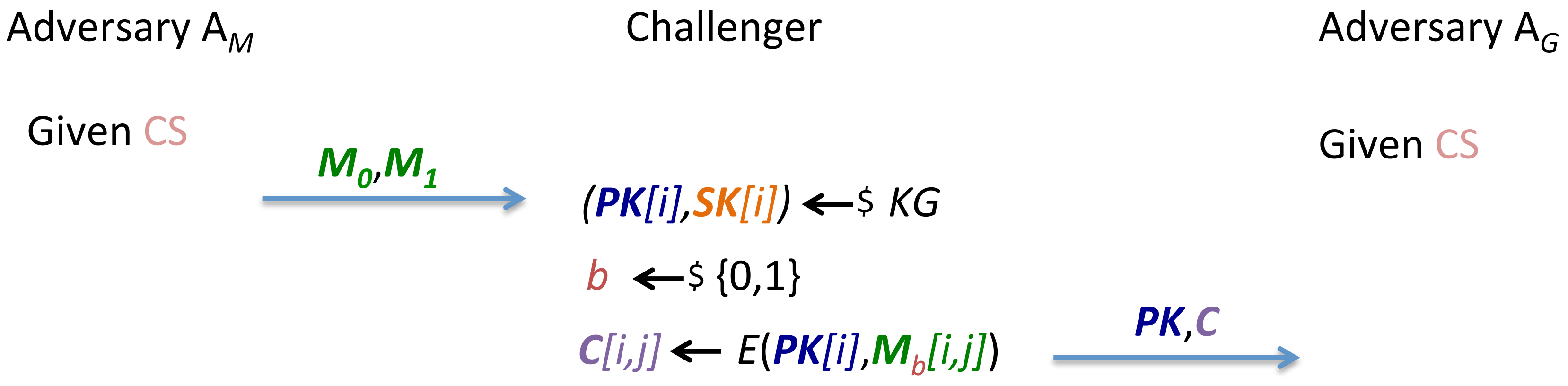
Adversary A_G

Given CS

Given CS

mIND Security

Adversary A_{CS}: generate common state CS



Adversary A_M

Given CS

M_0, M_1

Challenger

$(PK[i], SK[i]) \leftarrow\$ \; KG$

$b \leftarrow\$ \; \{0,1\}$

$C[i,j] \leftarrow E(PK[i], M_b[i,j])$

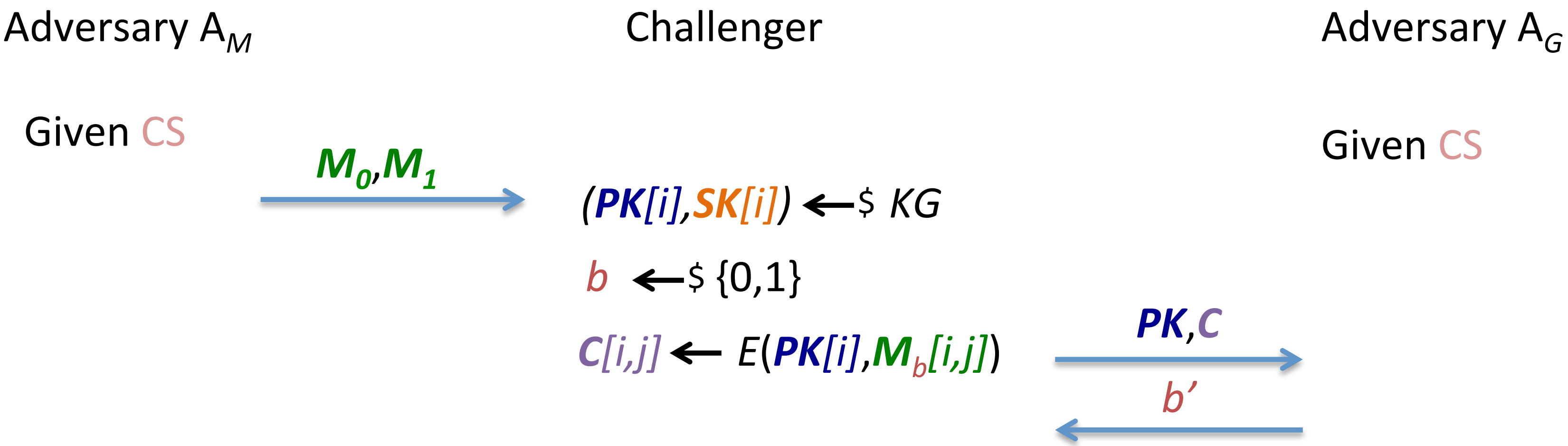
PK, C

Adversary A_G

Given CS

mIND Security

Adversary A_{CS}: generate common state CS



Adversary A_M

Given CS

M_0,M_1

Challenger

$(PK[i],SK[i]) \leftarrow\$ \ KG$

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

$C[i,j] \leftarrow E(PK[i],M_b[i,j])$

Adversary A_G

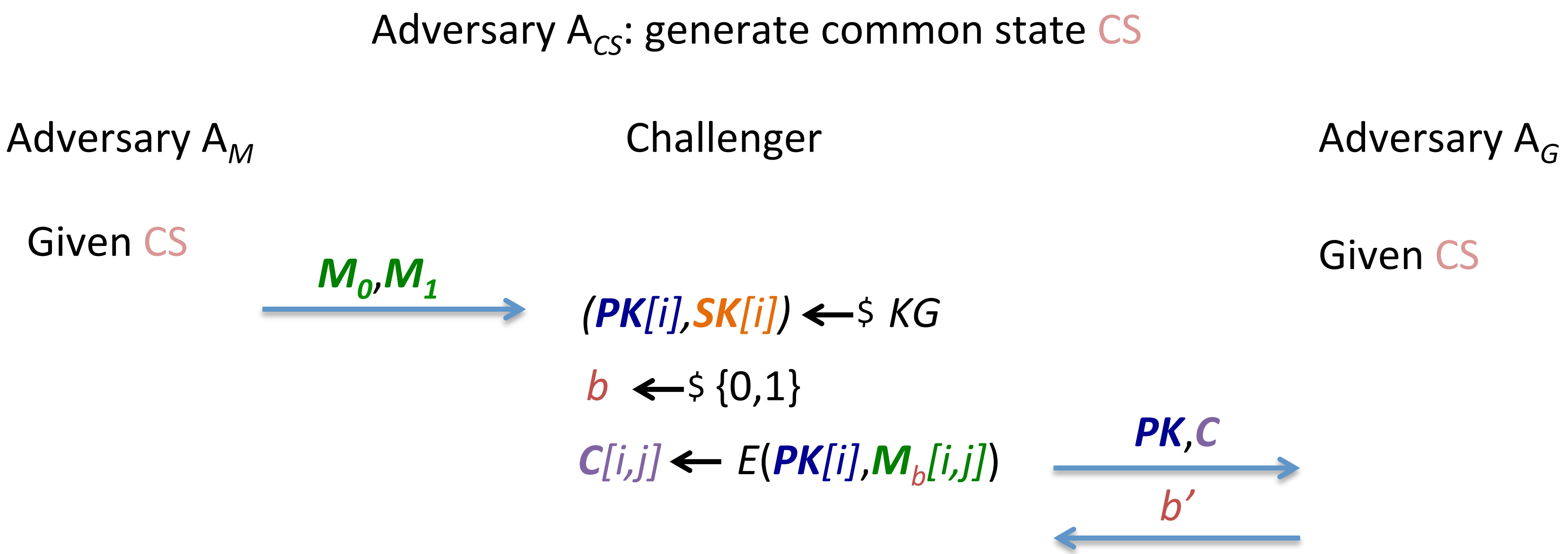
Given CS

PK,C

b'

Adversary (A_{CS},A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

mIND Security



Adversary A_{CS}: generate common state CS

Adversary A_M

Given CS

M_0,M_1

Challenger

$(PK[i],SK[i]) \leftarrow\$ KG$

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

$C[i,j] \leftarrow E(PK[i],M_b[i,j])$

PK,C

b'

Adversary A_G

Given CS

Adversary (A_{CS},A_M,A_G) wins if $b'=b$. Security requires $2\Pr[b'=b]-1$ to be negligible.

There are three essential restrictions without which security is not achievable:

1) A_{CS} and A_M do not get the public keys.

2) All messages in M_0,M_1 must have high min-entropy and there are no repeated messages in the same row of the matrix.

3) A_M cannot pass state to A_G.

Result

For randomized PKE, single-user security implies multi-user security [BBM00,BPS00]. What about deterministic PKE?

Result

For randomized PKE, single-user security implies multi-user security [BBM00,BPS00]. What about deterministic PKE?

It was conjectured by Bellare, Boldyreva and O’neill [BBO07] that single-user security does not imply multi-user security for deterministic PKEs.

Result

For randomized PKE, single-user security implies multi-user security [BBM00,BPS00]. What about deterministic PKE?

It was conjectured by Bellare, Boldyreva and O'neill [BBO07] that single-user security does not imply multi-user security for deterministic PKEs.

Theorem: Assume there exists an IND-secure deterministic PKE scheme. Then there exists a deterministic PKE scheme that is IND-secure, but not mIND-secure.

Result

For randomized PKE, single-user security implies multi-user security [BBM00,BPS00]. What about deterministic PKE?

It was conjectured by Bellare, Boldyreva and O'neill [BBO07] that single-user security does not imply multi-user security for deterministic PKEs.

Theorem: Assume there exists an IND-secure deterministic PKE scheme. Then there exists a deterministic PKE scheme that is IND-secure, but not mIND-secure.

Insecure even for two users.

Proof Idea

Case 1: All deterministic PKE schemes are insecure for 2 users.

Proof Idea

Case 1: All deterministic PKE schemes are insecure for 2 users.

Trivial to establish the theorem.

Proof Idea

Case 1: All deterministic PKE schemes are insecure for 2 users.

Trivial to establish the theorem.

Case 2: There exists a deterministic PKE scheme that is secure for 2 users.

Proof Idea

Case 1: All deterministic PKE schemes are insecure for 2 users.

Trivial to establish the theorem.

Case 2: There exists a deterministic PKE scheme that is secure for 2 users.

Let *DE* be a scheme which is secure for 2 users. Then we construct a modified scheme *DE’* which is secure for a single user, but not for 2 users.

Proof Idea

```
$DE'.PG(1^{\lambda})$:
    $\pi \xleftarrow{\$} DE.PG(1^{\lambda})$
    $(PK^*, SK^*) \xleftarrow{\$} DE.KG(\pi)$
    Return $\pi^* = (\pi, PK^*)$
```

Proof Idea

```
$DE'.PG(1^\lambda)$:
$\pi \leftarrow\$ \ DE.PG(1^\lambda)$
$(PK^*, SK^*) \leftarrow\$ \ DE.KG(\pi)$
Return $\pi^* = (\pi, PK^*)$
```

```
$DE'.KG(\pi^*)$:
$(PK, SK) \leftarrow\$ \ DE.KG(\pi)$
Return $(PK, SK)$
```

Proof Idea

```
$DE'.PG(1^{\lambda})$:
    $\pi \xleftarrow{\$} DE.PG(1^{\lambda})$
    $(PK^*, SK^*) \xleftarrow{\$} DE.KG(\pi)$
    Return $\pi^* = (\pi, PK^*)$
```

```
$DE'.KG(\pi^*)$:
    $(PK, SK) \xleftarrow{\$} DE.KG(\pi)$
    Return $(PK, SK)$
```

```
$DE'.E(\pi^*, PK, M)$:
    $C \leftarrow DE.E(\pi, PK, M)$
    $C^* \leftarrow DE.E(\pi, PK^*, M)$
    Return $C' = (C, C^*)$
```

Proof Idea

```
$DE'.PG(1^\lambda)$:
    $\pi \leftarrow\$ \; DE.PG(1^\lambda)$
    $(PK^*, SK^*) \leftarrow\$ \; DE.KG(\pi)$
    Return $\pi^* = (\pi, PK^*)$
```

```
$DE'.KG(\pi^*)$:
    $(PK, SK) \leftarrow\$ \; DE.KG(\pi)$
    Return $(PK, SK)$
```

```
$DE'.E(\pi^*, PK, M)$:
    $C \leftarrow DE.E(\pi, PK, M)$
    $C^* \leftarrow DE.E(\pi, PK^*, M)$
    Return $C' = (C, C^*)$
```

```
$DE'.D(\pi^*, SK, C')$:
    $M \leftarrow DE.D(\pi, SK, C)$
    Return $M$
```

Proof Idea

```
$DE'.PG(1^\lambda)$:
$\pi \leftarrow_\$ DE.PG(1^\lambda)$
$(PK^*, SK^*) \leftarrow_\$ DE.KG(\pi)$
Return $\pi^* = (\pi, PK^*)$
```

```
$DE'.KG(\pi^*)$:
$(PK, SK) \leftarrow_\$ DE.KG(\pi)$
Return $(PK, SK)$
```

```
$DE'.E(\pi^*, PK, M)$:
$C \leftarrow DE.E(\pi, PK, M)$
$C^* \leftarrow DE.E(\pi, PK^*, M)$
Return $C' = (C, C^*)$
```

```
$DE'.D(\pi^*, SK, C')$:
$M \leftarrow DE.D(\pi, SK, C)$
Return $M$
```

PK^* can be viewed as a key of a dummy second user of the old scheme.

Proof Idea

```
$DE'.PG(1^\lambda)$:
$\pi \xleftarrow{\$} DE.PG(1^\lambda)$
$(PK^*, SK^*) \xleftarrow{\$} DE.KG(\pi)$
Return $\pi^* = (\pi, PK^*)$
```

```
$DE'.KG(\pi^*)$:
$(PK, SK) \xleftarrow{\$} DE.KG(\pi)$
Return $(PK, SK)$
```

```
$DE'.E(\pi^*, PK, M)$:
$C \leftarrow DE.E(\pi, PK, M)$
$C^* \leftarrow DE.E(\pi, PK^*, M)$
Return $C' = (C, C^*)$
```

```
$DE'.D(\pi^*, SK, C')$:
$M \leftarrow DE.D(\pi, SK, C)$
Return $M$
```

PK^* can be viewed as a key of a dummy second user of the old scheme.

Then the fact that DE' is IND-secure follows from the security against 2 users of the original scheme.

Proof Idea

```
$DE'.PG(1^{\lambda})$:
$\pi \leftarrow\$ \ DE.PG(1^{\lambda})$
$(PK^*, SK^*) \leftarrow\$ \ DE.KG(\pi)$
Return $\pi^* = (\pi, PK^*)$
```

```
$DE'.KG(\pi^*)$:
$(PK, SK) \leftarrow\$ \ DE.KG(\pi)$
Return $(PK, SK)$
```

```
$DE'.E(\pi^*, PK, M)$:
$C \leftarrow DE.E(\pi, PK, M)$
$C^* \leftarrow DE.E(\pi, PK^*, M)$
Return $C' = (C, C^*)$
```

```
$DE'.D(\pi^*, SK, C')$:
$M \leftarrow DE.D(\pi, SK, C)$
Return $M$
```

The fact that *DE′* is not secure for 2 users follows from the fact that the second part of the ciphertexts can be used to check whether the messages encrypted to different users are the same or not.

Summary

- Consider using the definition with three stage adversaries. For the one with two stage adversaries it is not clear whether security in the standard model implies security in the random oracle model.

Summary

- Consider using the definition with three stage adversaries. For the one with two stage adversaries it is not clear whether security in the standard model implies security in the random oracle model.

- It is not possible to obtain deterministic PKE which are secure against selective opening attacks (at least for simulation-based definitions).

Summary

✧ Consider using the definition with three stage adversaries. For the one with two stage adversaries it is not clear whether security in the standard model implies security in the random oracle model.

✧ It is not possible to obtain deterministic PKE which are secure against selective opening attacks (at least for simulation-based definitions).

✧ Single-user security does not imply multi-user security for deterministic PKE.

Thank You!