CS489/689
Privacy, Cryptography, Network and Data Security

Winter 2023, Tuesday/Thursday 8:30-9:50am

Today

- Recap: security games
- El gamal cryptosystem
- El gamal signatures
- El gamal security
- Crash course mathematics: spliced in some terminology/concepts

What on earth are groups…

Groups - Basically a set with specific properties

Def: A group is a set with an operation on its elements which:

- Is closed
- Has an identity
- Is associative,
- And every element has an inverse

Closed - With Addition as the operation

For every a,b in **Z/NZ**: a+b in **Z/NZ**

Aka:

The sum of two group elements is an element in the group.

Has an Identity: With Addition as the operation

E.g., $a+0 = a$

Has an element e such that any element plus e outputs the element (itself)

Is Associative: With Addition as the operation

$(a+b)+c = a + (b+c)$

Every element has an inverse

Integers, additive inverse of a is -a

$a + (-a) = (-a) + a = 0$

Abelian Groups

Def: Abelian groups are groups which are commutative.

The property: applying the group operation to two group elements does not depend on the order in which they are written.

E.g. $a+b = b+a$

**really useful in crypto, and is why we almost always use them

Decisional Diffie-Hellman

Crash Course: Decision Diffie-Hellman Problem

The adversary is given $g \in G$, $a=g^x$, $b=g^y$, and $c=g^z$, for unknowns x, y, and z.

Crash Course: Decision Diffie-Hellman Problem

The adversary is given $g \in G$, $a=g^x$, $b=g^y$, and $c=g^z$, for unknowns x, y, and z.

- Challenger chooses *z* s.t. *z=x*y (with pr=½)* or *z* is random
- **Goal** of adversary is to determine whether:

z=x*y **OR** random *z*

Crash Course: Decision Diffie-Hellman Problem

The adversary is given $g \in G$, $a=g^x$, $b=g^y$, and $c=g^z$, for unknowns x, y, and z.

- Challenger chooses *z* s.t. *z=x*y (with pr=½)* or *z* is random 
- **Goal** of adversary is to determine whether:

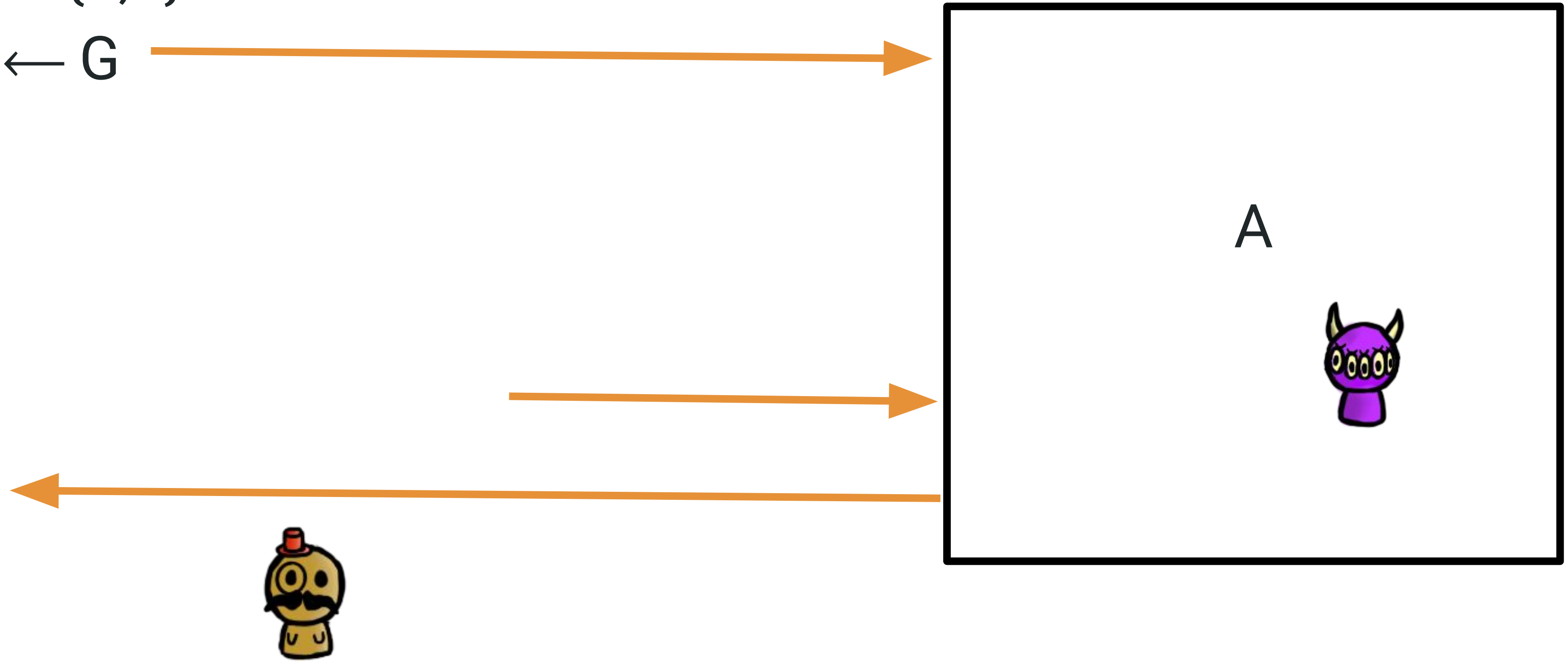
z=x*y  random *z*

$\mathbf{Adv}_G^{\mathbf{DDH}}\mathbf{(A)}$ = 2*|Pr[A wins the DDH game in G]-½|.

DDH Security Game

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

$g \leftarrow G$



DDH Security Game

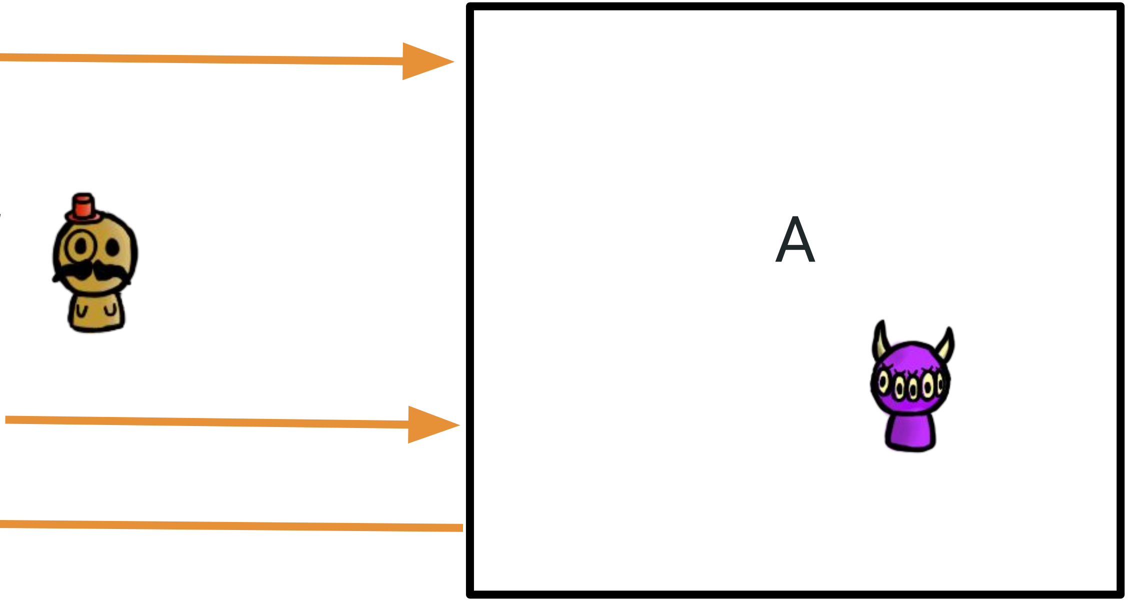
b ← {0,1}

g ← G

x,y ← ***Z****/q****Z***

If b=0 then *z* ← ***Z****/q****Z***

If b=1 then *z* ← x*y



DDH Security Game

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

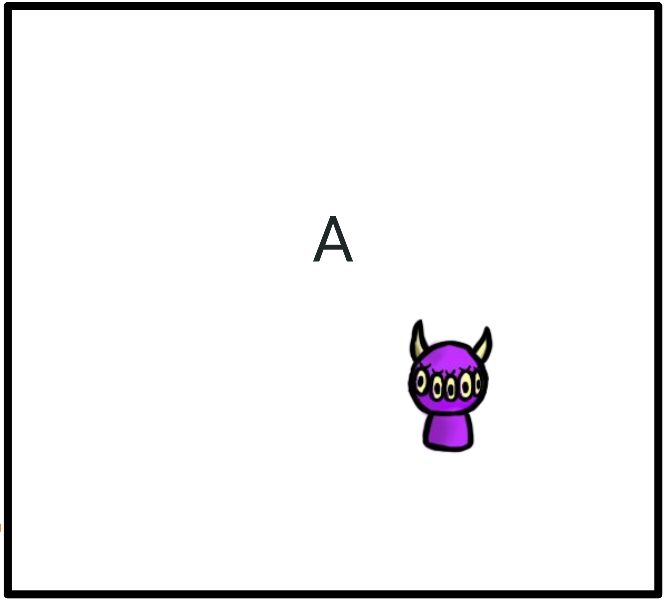
$g \leftarrow G$

$x,y \leftarrow \mathbb{Z}/q\mathbb{Z}$

If $b=0$ then $z \leftarrow \mathbb{Z}/q\mathbb{Z}$

If $b=1$ then $z \leftarrow x*y$

$a \leftarrow g^x$, $b \leftarrow g^y$, $c \leftarrow g^z$



DDH Security Game

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

$g \leftarrow G$

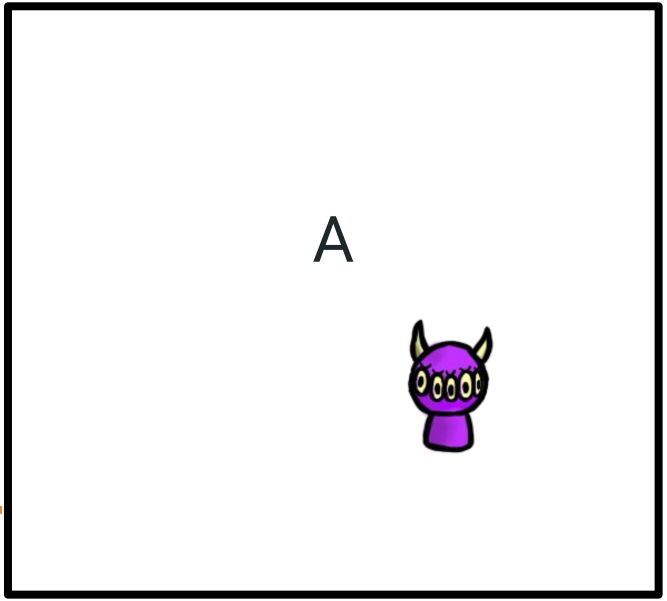
$x,y \leftarrow \boldsymbol{Z}/q\boldsymbol{Z}$

If $b=0$ then $z \leftarrow \boldsymbol{Z}/q\boldsymbol{Z}$

If $b=1$ then $z \leftarrow x*y$

$a \leftarrow g^x$, $b \leftarrow g^y$, $c \leftarrow g^z$

b'



Win if $b'=b$

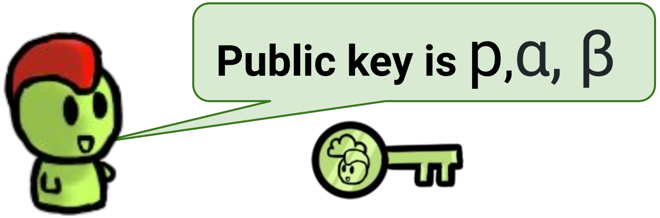
$\mathbf{Adv_G^{DDH}(A)}$ = 2*|Pr[A wins the DDH game in G]-½|.

El Gamal

- **1985 by Taher ElGamal**

ElGamal Public Key Cryptosystem

- Let p be a prime such that the DLP in $(\mathbf{Z}_p^*, \cdot)$ is infeasible
- Let $\alpha \in \mathbf{Z}_p^*$ be a primitive element
- Let $\mathcal{P} = \mathbf{Z}_p^*$, $\mathcal{C} = \mathbf{Z}_p^* \times \mathbf{Z}_p^*$ and…
- $\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\}$
- For a secret random number k in $\mathbf{Z}_{p-1}$ define:
  - $e_K(x,k) = (y_1, y_2)$, where $y_1 = \alpha^k \bmod p$ and $y_2 = x\beta^k \bmod p$
- For y_1, y_2 in $\mathbf{Z}_p^*$, define $d_K(y_1, y_2) = y_2(y_1^{a})^{-1} \bmod p$



ElGamal: The Keys

1. Bob picks a “large” prime p and a primitive root α.
   a. Assume message m is an integer $0 < m < p$
2. Bob picks secret integer a
3. Bob Computes $\beta \equiv \alpha^a \pmod{p}$

ElGamal: The Keys

1. Bob picks a “large” prime p and a primitive root α.
    a. Assume message m is an integer $0 < m < o$

2. Bob picks secret integer a
3. Bob Computes $\beta \equiv \alpha^a \pmod{p}$
4. Bob’s public key is (p, α, β) 

ElGamal: The Keys

1. Bob picks a “large” prime p and a primitive root α.
    a. Assume message m is an integer $0 < m < o$

2. Bob picks secret integer a
3. Bob Computes $\beta \equiv \alpha^a \pmod{p}$
4. Bob’s public key is (p, α, β) 
5. Bob’s private key is a 

ElGamal: Encryption

I choose secret integer k

ElGamal: Encryption

I choose secret integer k

Compute $y_1 \equiv \alpha^k \pmod{p}$

ElGamal: Encryption

I choose secret integer k

Compute $y_1 \equiv \alpha^k \pmod{p}$

Compute $y_2 \equiv \beta^k m \pmod{p}$

ElGamal: Encryption

I choose secret integer k

Compute $y_1 \equiv \alpha^k \pmod{p}$

Compute $y_2 \equiv \beta^k m \pmod{p}$

ElGamal: Encryption

I choose secret integer k

Compute $y_1 \equiv \alpha^k \pmod{p}$

Compute $y_2 \equiv \beta^k m \pmod{p}$

Send y_1 and y_2 to Bob

ElGamal: Decryption

I choose secret integer k

Compute $y_1 \equiv \alpha^k \pmod{p}$

Compute $y_2 \equiv \beta^k m \pmod{p}$

Send y_1 and y_2 to Bob

Compute $y_1 y_2^{-a} \equiv m \pmod{p}$

ElGamal: Decryption

I choose secret integer k

Compute $y_1 \equiv \alpha^k \pmod{p}$

Compute $y_2 \equiv \beta^k m \pmod{p}$

Send y_1 and y_2 to Bob

Compute $y_2 y_1^{-a} \equiv m \pmod{p}$

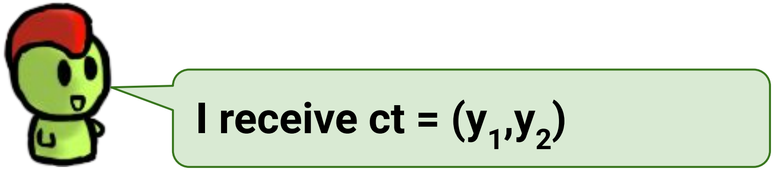
This works because: $y_2 y_1^{-a} \equiv \beta^k m (\alpha^k)^{-a} \equiv m \pmod{p}$

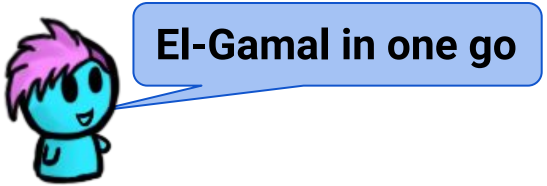
ElGamal Informal Summary

- The plaintext m is “hidden” by multiplying it by β^k to get y_2

ElGamal Informal Summary

- The plaintext m is “hidden” by multiplying it by β^k to get y_2
- The ciphertext includes α^k so that Bob can compute β^k from α^k (because Bob knows a)

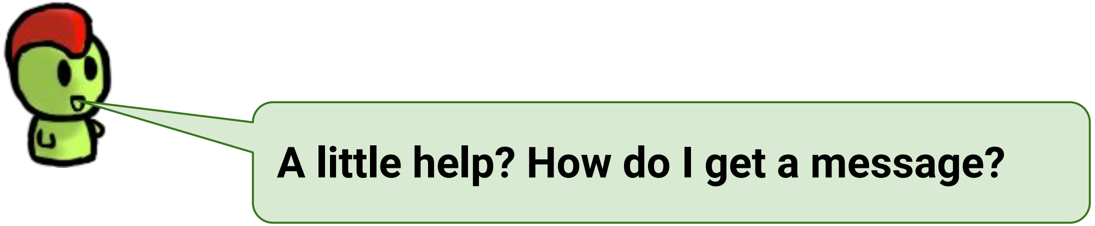




ElGamal Informal Summary

- The plaintext x is “hidden” by multiplying it by β^k to get y_2
- The ciphertext includes α^k so that Bob can compute β^k from α^k (because Bob knows a)
- Thus, Bob can “reveal” m by dividing y_2 by β^k

Example: How ElGamal works

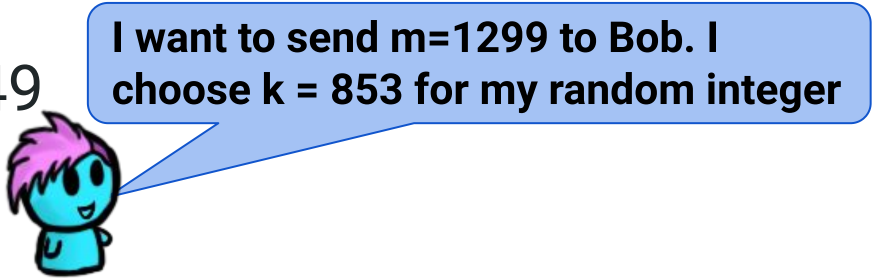


Example: How El Gamal works

- Set p=2579 and α = 2 (α is a primitive element modulo p) and let a =765, then
- $\beta = 2^{765} \bmod 2579 = 949$

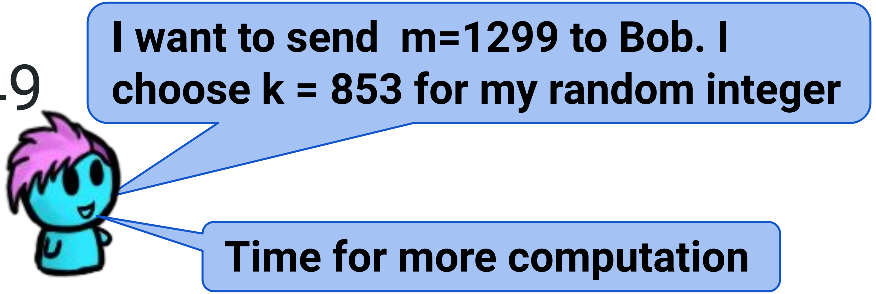
Example: How El Gamal works

- Set p=2579 and α = 2 (α is a primitive element modulo p) and let a =765, then
- $\beta = 2^{765} \bmod 2579 = 949$



Example: How El Gamal works

- Set p=2579 and α = 2 (α is a primitive element modulo p) and let a =765, then
- $\beta = 2^{765} \bmod 2579 = 949$



Example: How El Gamal works

- Set p=2579 and α = 2 (α is a primitive element modulo p) and let a =765, then
- $\beta = 2^{765} \bmod 2579 = 949$

- $y_1 = 2^{853} \bmod 2579 = 435$, and
- $y_2 = 1299 * 949^{853} \bmod 2579 = 2396$

Example: How ElGamal works

- Ok, we have y_1 and y_2
- $y_1 = 2^{853} \bmod 2579 = 435$, and
- $y_2 = 1299 * 949^{853} \bmod 2579 = 2396$

I receive ct = y = (435,2396)

Example: How ElGamal works

- $y_1 = 2^{853} \bmod 2579 = 435$, and
- $y_2 = 1299 * 949^{853} \bmod 2579 = 2396$

- $m = 2396 * (435^{765})^{-1} \bmod 2579 = 1299$

Example: How ElGamal works

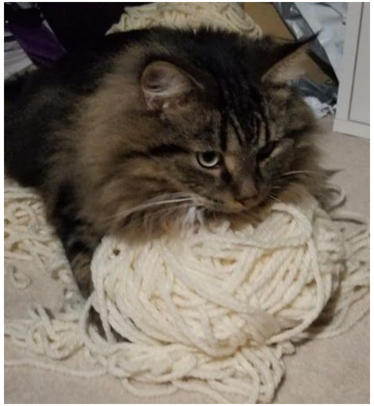
- $y_1 = 2^{853} \bmod 2579 = 435$, and
- $y_2 = 1299 * 949^{853} \bmod 2579 = 2396$

- $m = 2396 * (435^{765})^{-1} \bmod 2759 = 1299$

ElGamal...Encrypt. “Small” Calculation Day

Act.

- $(p, \alpha, \beta) = (809, 256, 498)$
- $a = 68$
- $k = 89$
- $m=100$



Determine $c = y_1, y_2$.

Submit c and a short description of your computation.

Security of El Gamal

El-Gamal$_{SIM}$ Relies on DDH

Given g, g^a, g^b distinguish a random r and g^{ab}

Known computationally hard problem

Short Answer?

- Let *p* be a prime such that the DLP in $(\mathbf{Z}_p^*, \cdot)$ is infeasible
- Let $\alpha \in \mathbf{Z}_p^*$ be a primitive element
- Let $\mathcal{P} = \mathbf{Z}_p^*$, $\mathcal{C} = \mathbf{Z}_p^* \times \mathbf{Z}_p^*$ and…
- $\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod p\}$
- For a secret random number k in $\mathbf{Z}_{p-1}$ define:
  - $e_K(x,k) = (y_1, y_2)$, where $y_1 = \alpha^k \bmod p$ and $y_2 = x\beta^k \bmod p$
- For y_1, y_2 in $\mathbf{Z}_p^*$, define $d_K(y_1, y_2) = y_2(y_1^{\,a})^{-1} \bmod p$

Clearly insecure if: Adversary can compute $a = \log_\alpha \beta$, then could decrypt the same as Bob.

Short Answer?

- Let p be a prime such that the DLP in $(\mathbf{Z}_p^*, .)$ is infeasible
- Let $\alpha \in \mathbf{Z}_p^*$ be a primitive element
- Let $\mathcal{P} = \mathbf{Z}_p^*$, $\mathcal{C} = \mathbf{Z}_p^* \times \mathbf{Z}_p^*$ and...
- $\mathcal{K}=\{(p, \alpha, a, \beta): \beta=\alpha^a \pmod p\}$

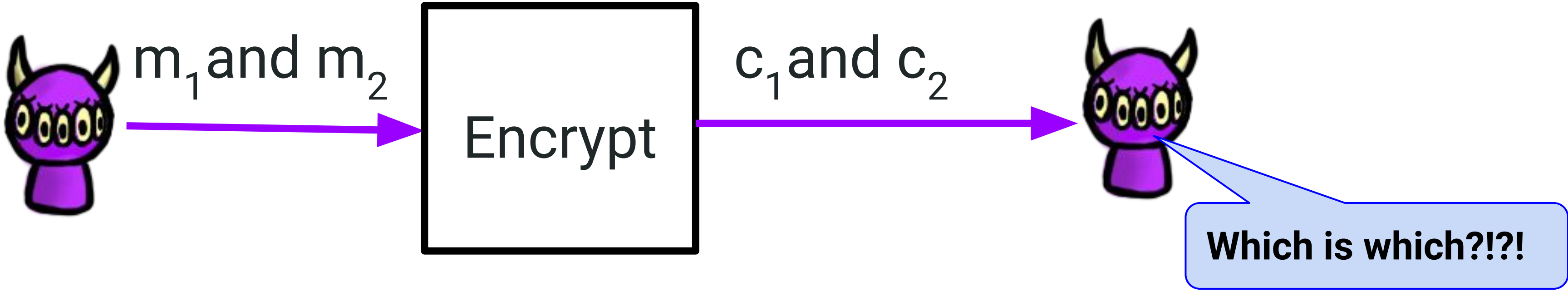
Necessary condition for security: DLP in $\mathbf{Z}_p^*$ is infeasible

  - $e_K(x,k) = (y_1, y_2)$, where $y_1 = \alpha^k \bmod p$ and $y_2 = x\beta^k \bmod p$
- For y_1, y_2 in $\mathbf{Z}_p^*$, define $d_K(y_1, y_2)=y_2(y_1^{a})\text{-}1 \bmod p$

Clearly insecure if: Adversary can compute $a=\log_\alpha\beta$, then could decrypt the same as Bob.

Recall: IND-CPA

IND-CPA secure: if a polynomial time adversary choosing two plaintexts **cannot distinguish** between the resulting ciphertexts.

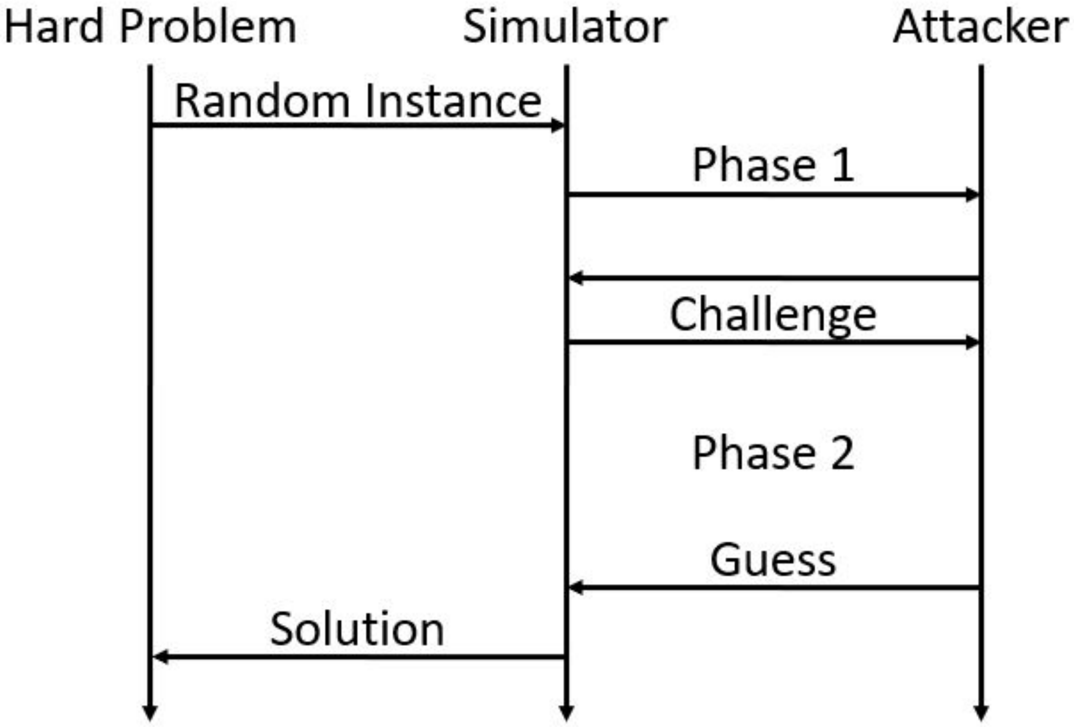


Proving IND-CPA Using Simulators

- The simulator is given an arbitrary instance of a known to be hard problem
- The simulator interacts with the attacker
- The simulator solves the hard problem, if the attacker is successful.

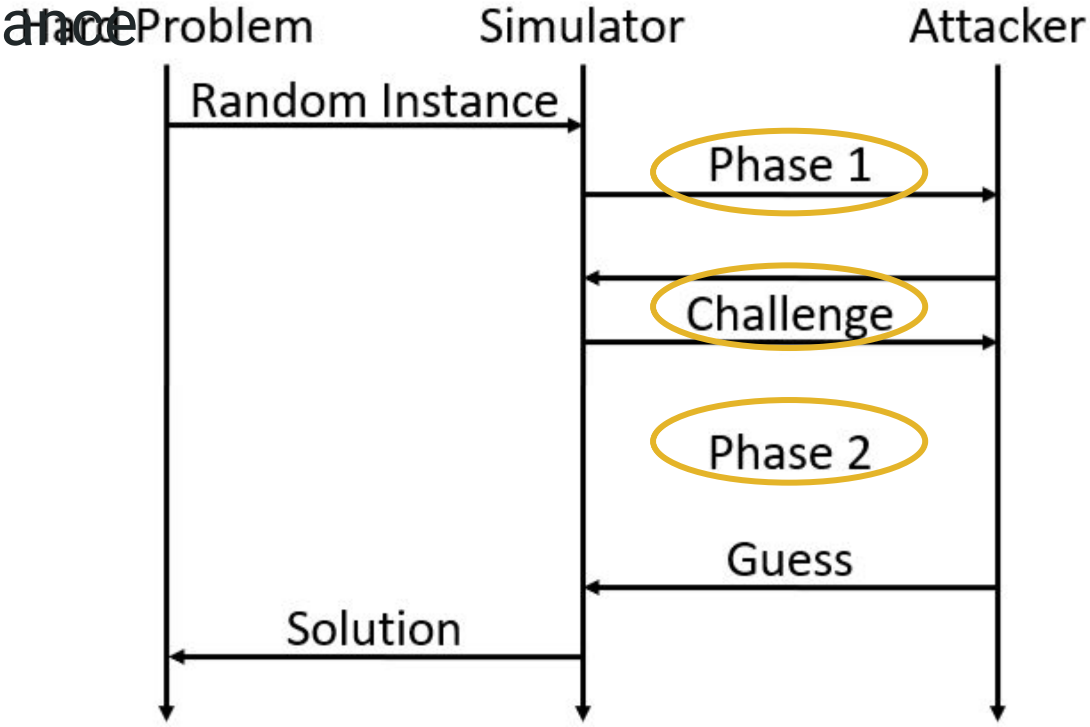
Think of the security games earlier.

Simulator Proofs…Wait What?



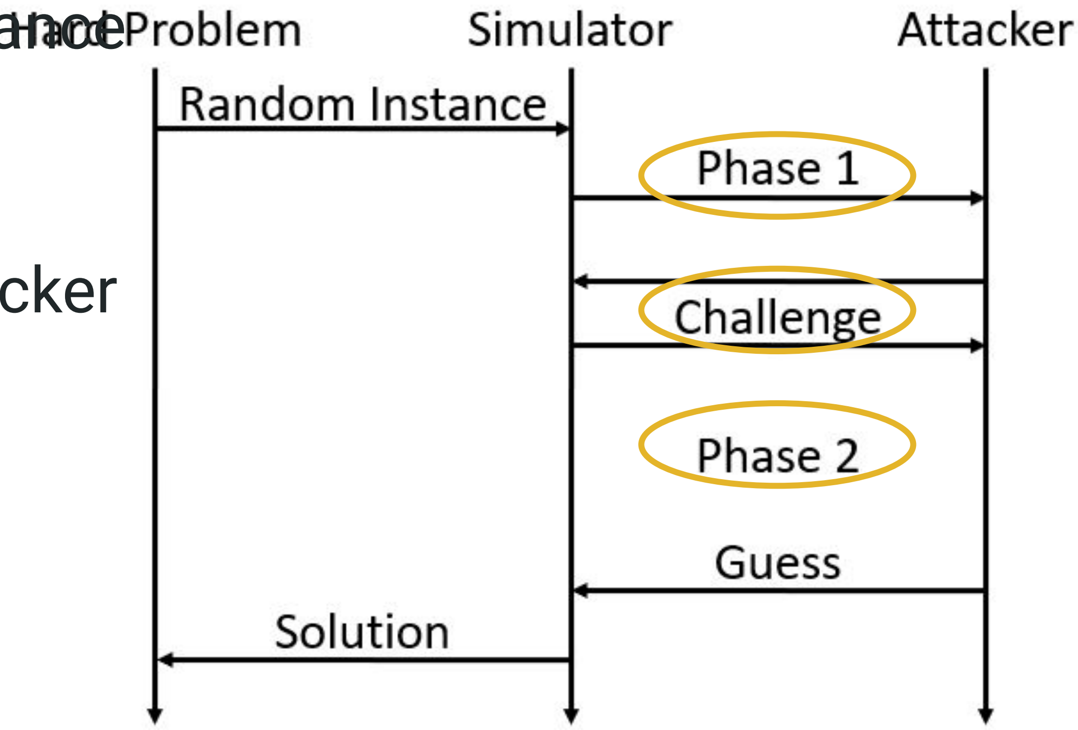
Simulator Proofs…Wait What?

- S receives arbitrary instance of known to be hard problem



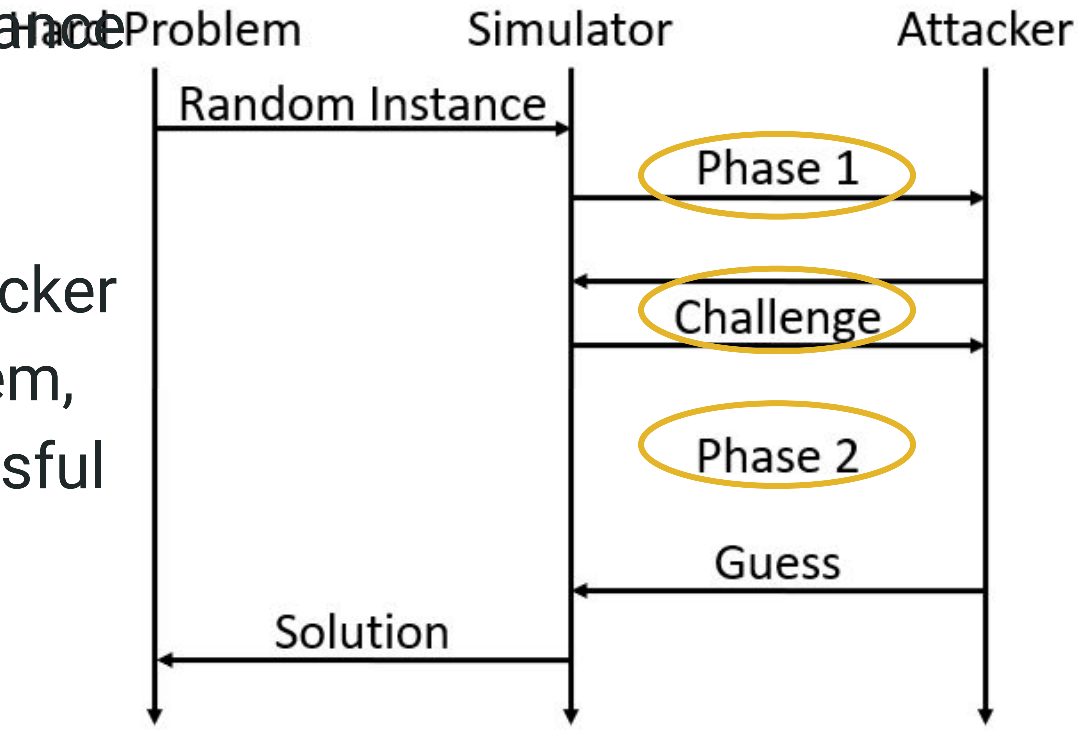
Simulator Proofs...Wait What?

- S receives arbitrary instance of known to be hard problem
- S interacts with the attacker



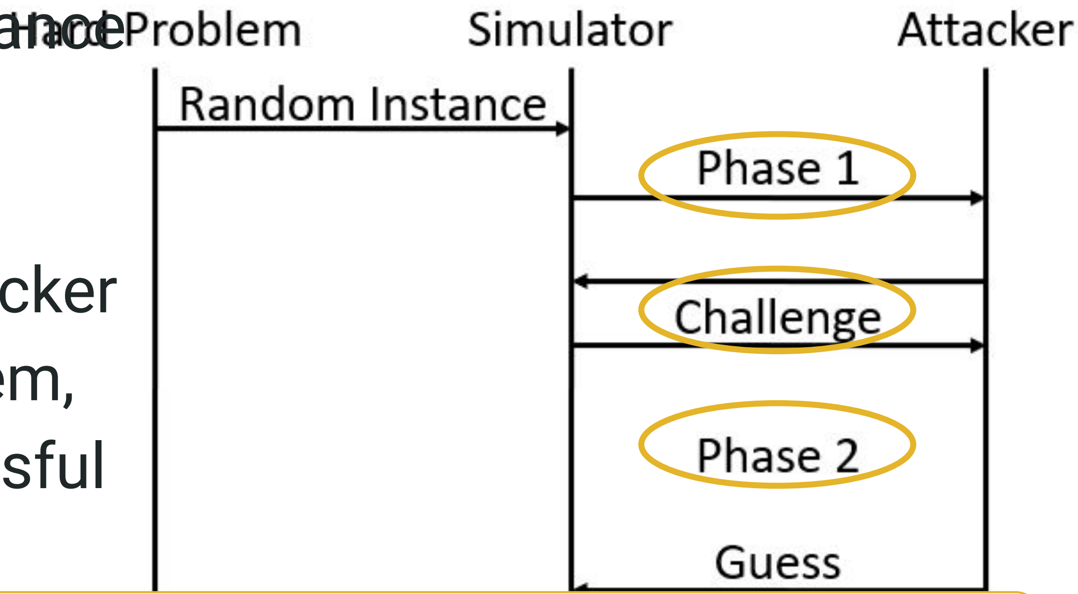
Simulator Proofs...Wait What?

- S receives arbitrary instance of known to be hard problem
- S interacts with the attacker
- S solves the hard problem, if the attacker is successful



Simulator Proofs…Wait What?

- S receives arbitrary instance of known to be hard problem
- S interacts with the attacker
- S solves the hard problem, if the attacker is successful



The system is at least as "secure" as the problem is hard.

Recall from earlier: DDH Security Game

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

$g \leftarrow G$

$x,y \leftarrow \boldsymbol{Z}/q\boldsymbol{Z}$

If $b=0$ then $z \leftarrow \boldsymbol{Z}/q\boldsymbol{Z}$

If $b=1$ then $z \leftarrow x*y$

$a \leftarrow g^x$, $b \leftarrow g^y$, $c \leftarrow g^z$

b'

Win if $b'=b$

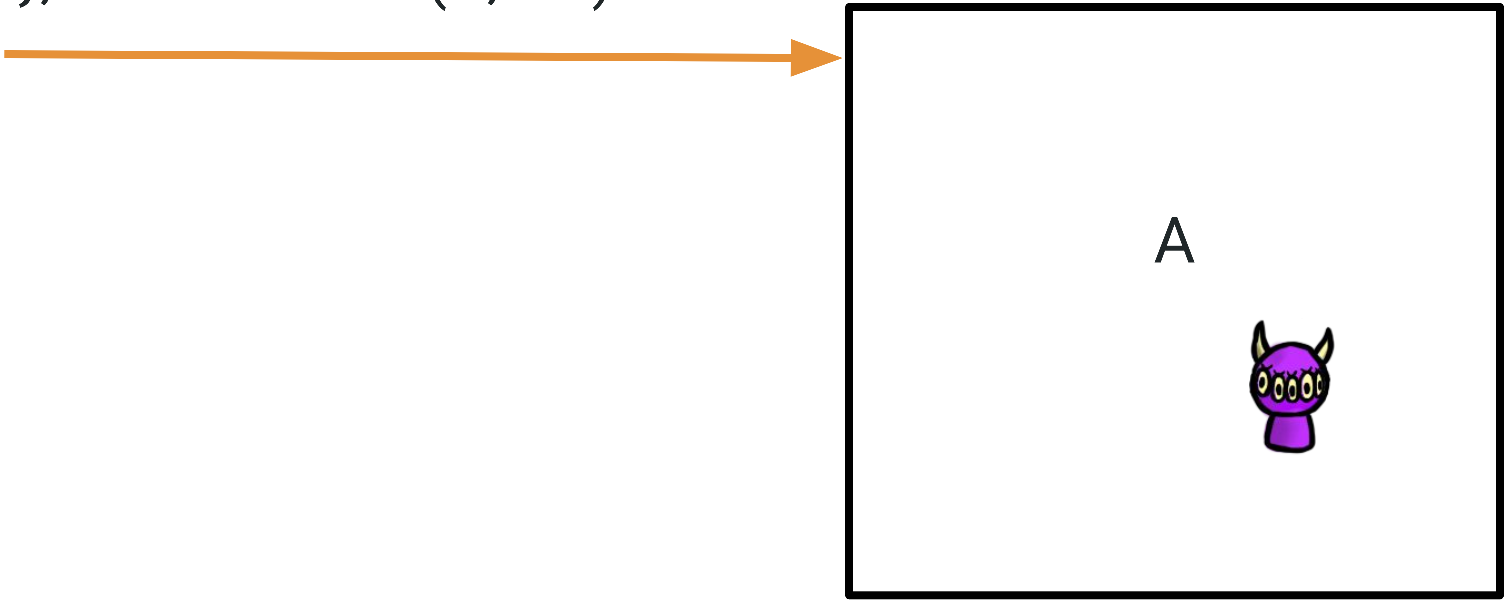
A

Adv_G^{DDH}(A) = 2*|Pr[A wins the DDH game in G]-½|.

El Gamal IND-CPA Game

b $\longleftarrow$ {0,1}, and random(K, K^{-1})

K

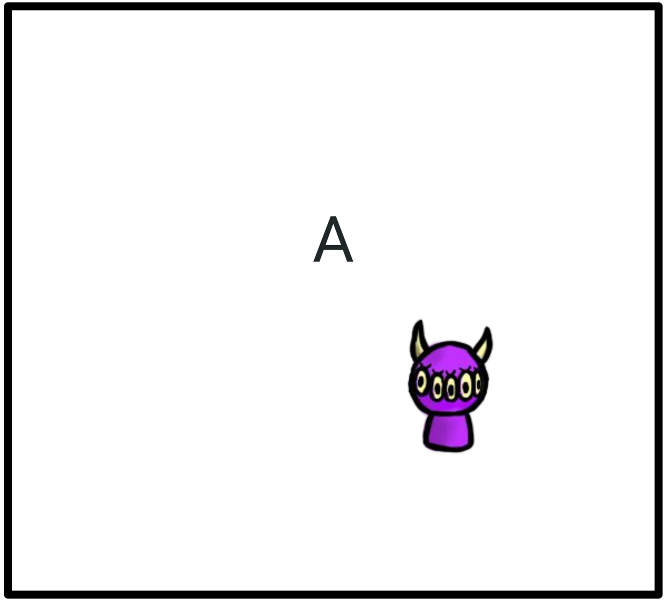


El Gamal IND-CPA Game

$b \leftarrow \{0,1\}$, and random(K, K^{-1})

K

M_0 and M_1 of equal length



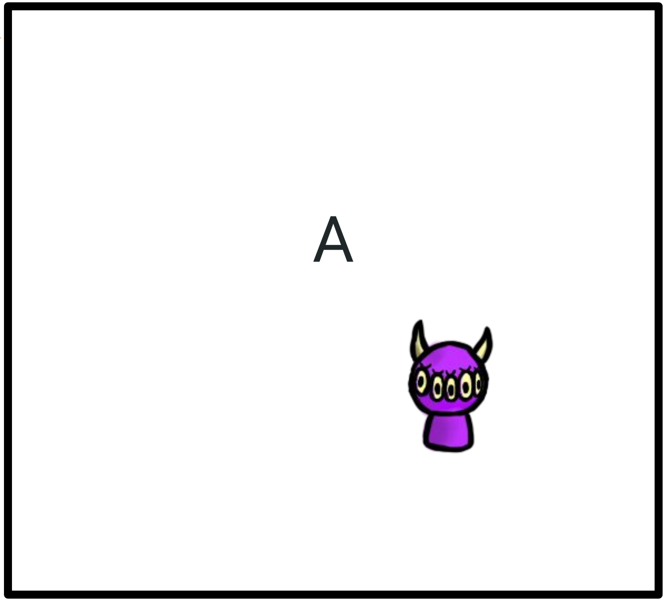
El Gamal IND-CPA Game

$b \leftarrow \{0,1\}$, and random(K, K^{-1})

K

M_0 and M_1 of equal length

$C = E_k[M_b]$



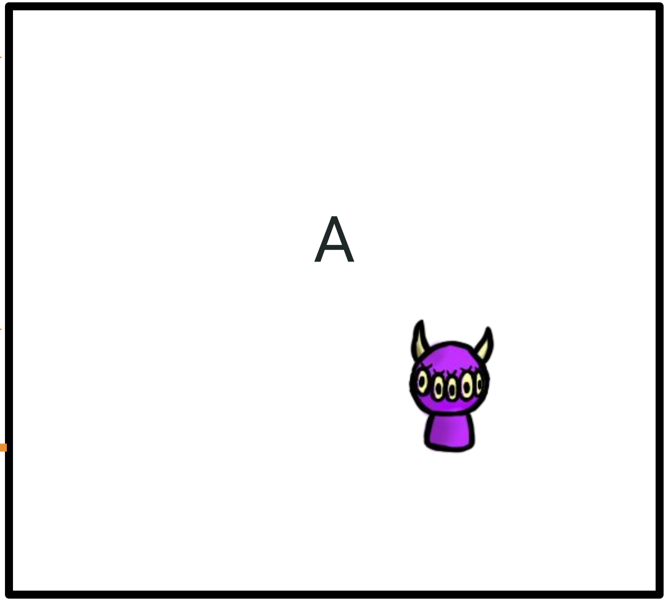
El Gamal IND-CPA Game

$b \leftarrow \{0,1\}$, and random(K, K^{-1})

K

M_0 and M_1 of equal length

$C = E_k[M_b]$



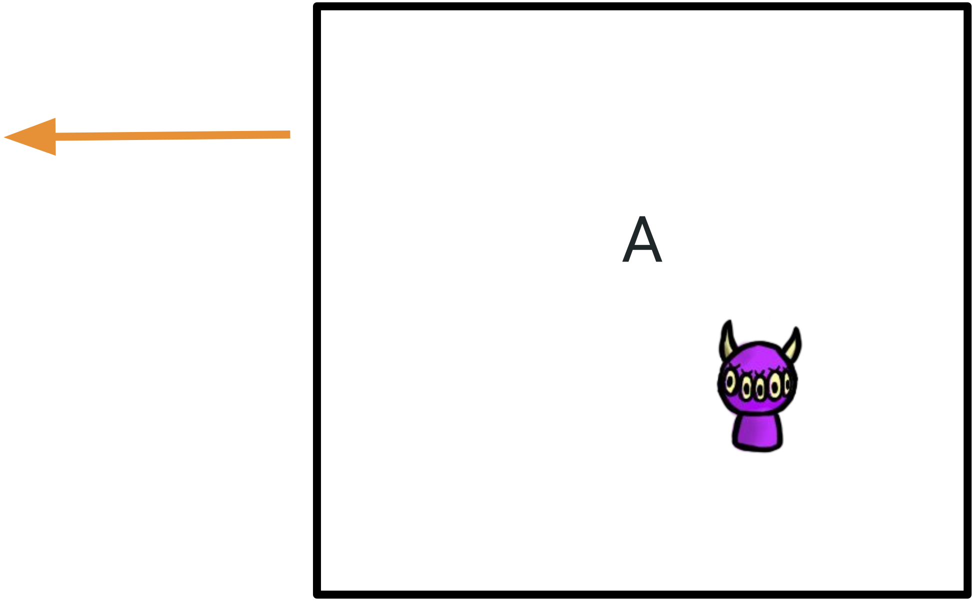
$b' \in \{0,1\}$

Attacker wins if $b=b'$

ElGamal Simulator IND-CP

g^a, g^c, g^d, and r

M_0 and M_1 of equal length



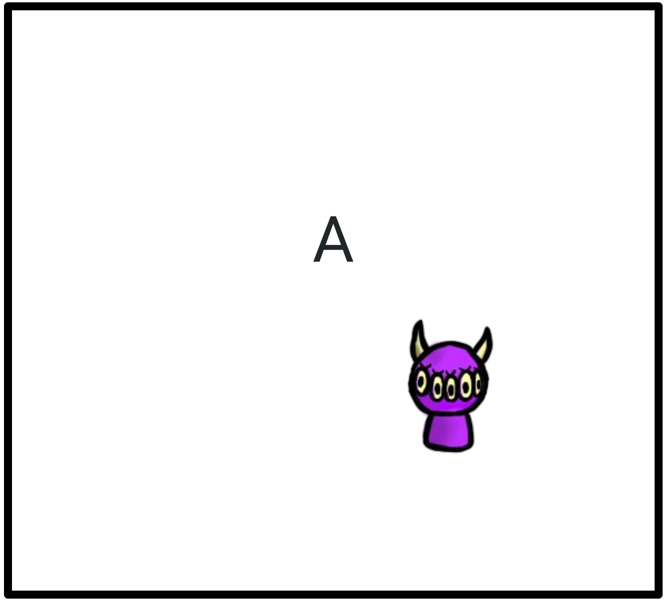
ElGamal Simulator IND-CP

g^a, g^c, g^d, and r

M_0 and M_1 of equal length

Set random r and b $\leftarrow$ {0,1}

Computed c_b



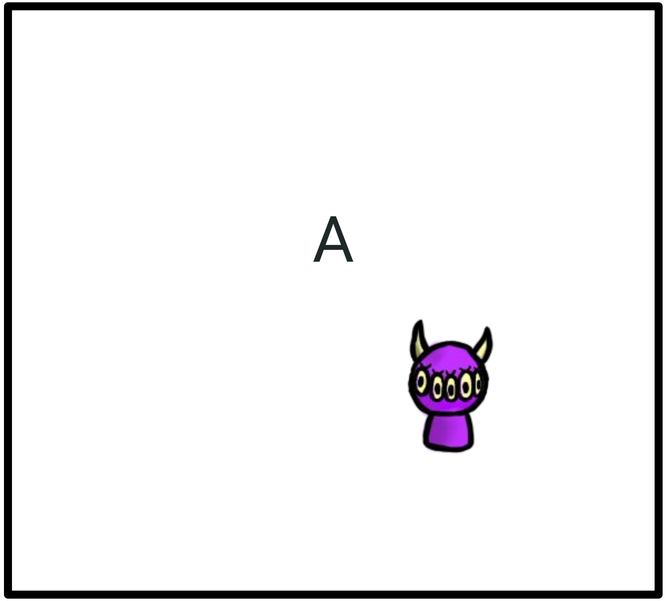
ElGamal Simulator IND-CP

g^a, g^c, g^d, and r

M_0 and M_1 of equal length

Set random r and b ⟵{0,1}

Computed c_b



Guess b’ for which M encrypted

Attacker wins if b=b’ , Output: $r=g^{ac}$

Network Security - Next week

Answer to activity...

- Ciphertext: $y_1 = 468$, $y_2 = 494$

Short Answer?

- Let *p* be a prime such that the DL… …feasible
- Let $\alpha \in \mathbf{Z}_p^*$ be a primitive…
- Let $\mathcal{P} = \mathbf{Z}_p^*$, $\mathcal{C} = \mathbf{Z}_p^*$ …
- $\mathcal{K}=\{(p, \alpha, a, \beta): \beta=$ …

a: must be secret, and must not be repeated

Necess… DLP in $\mathbf{Z}_p^*$ is infeasible

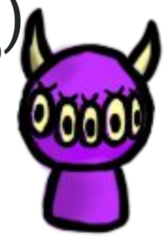
  - $e_K(x,k) = (y_1, y_2)$, where $y_1 =$ … mod p and $y_2 = x\beta^k$ mod p
- For y_1, y_2 in $\mathbf{Z}_p^*$, define $d_K(y_1, y_2)=y_2(y_1{}^a)$-1 mod p

Clearly insecure if: Adversary can compute $a=\log_\alpha\beta$, then could decrypt the same as Bob.

Repeating Private “a” in ElGamal

What if i reuse a for two messages m_a and m_b

- Then the ciphertexts are (y_1, y_{2a}) and (y_1, y_{2b})
- If Eve learns m_a, then she can learn m_b
- Eve computes:



$$-\mathbf{y_{2a}/m_a} \equiv \beta^k \equiv y_{2b}/m_b \pmod{p} \Rightarrow m_b \equiv \mathbf{(y_{2b}m_a)/y_{2a}}$$