Cryptography Primer

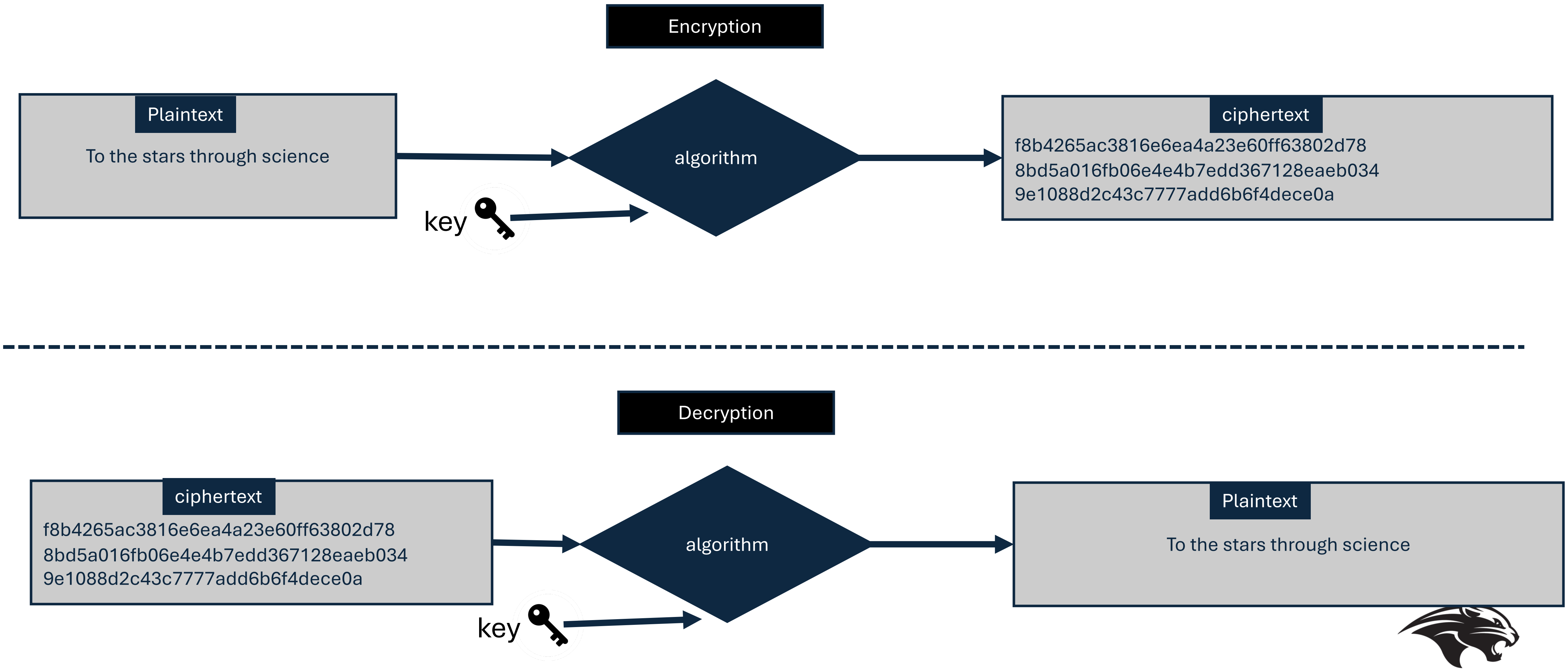
References

- https://www.cryptool.org/en/
- Cryptography E-Mates [Link]

Objectives

- Identify the components of a cryptographic systems
- Examine a one-time pad to understand good properties of cryptographic systems.
- Explore a method for exchange secure keys in a cryptographic system.

Cryptographic System



One-Time Pad Cryptographic System

T	O	T	H	E	S	T	A	R	S	PLAINTEXT
84	79	84	72	69	83	84	65	82	83	DECIMAL

11	22	33	44	55	9	8	7	6	5	KEY

CIPHERTEXT = XOR (KEY, PLAINTEXT)

11 xor 84	22 xor 79	33 xor 84	44 xor 72	55 xor 69	9 xor 83	8 xor 84	7 xor 65	6 xor 82	5 xor 83	
95	89	117	100	114	90	92	70	84	86	CIPHERTEXT

For a long enough key space, a one-time pad is a perfectly secure cryptographic system.

Cryptographic System Properties

A	A	A	A	B	B	B	B	B	C	PLAINTEXT
65	65	65	65	66	66	66	66	66	67	DECIMAL

11	22	33	44	55	9	8	7	6	5	KEY

CIPHERTEXT = XOR (KEY, PLAINTEXT)

11 xor 65	22 xor 65	33 xor 65	44 xor 65	55 xor 66	9 xor 66	8 xor 66	7 xor 66	6 xor 66	5 xor 67	
74	87	96	109	118	75	74	69	68	70	CIPHERTEXT

Lets examine an example encoding the plaintext AAAABBBC to understand some cryptographic properties

Two of the same inputs yield different outputs

A	A	A	A	B	B	B	B	B	C	PLAINTEXT
65	65	65	65	66	66	66	66	66	67	DECIMAL

11	22	33	44	55	9	8	7	6	5	KEY

CIPHERTEXT = XOR (KEY, PLAINTEXT)

11 xor 65	22 xor 65	33 xor 65	44 xor 65	55 xor 66	9 xor 66	8 xor 66	7 xor 66	6 xor 66	5 xor 67	
74	87	96	109	118	75	74	69	68	70	CIPHERTEXT

Notice how the first two As yield different results.
XOR(65,11) = 74
XOR(65,22) = 87

Two different inputs yield the same output

A	A	A	A	B	B	B	B	B	C	PLAINTEXT
65	65	65	65	66	66	66	66	66	67	DECIMAL

11	22	33	44	55	9	8	7	6	5	KEY

CIPHERTEXT = XOR (KEY, PLAINTEXT)

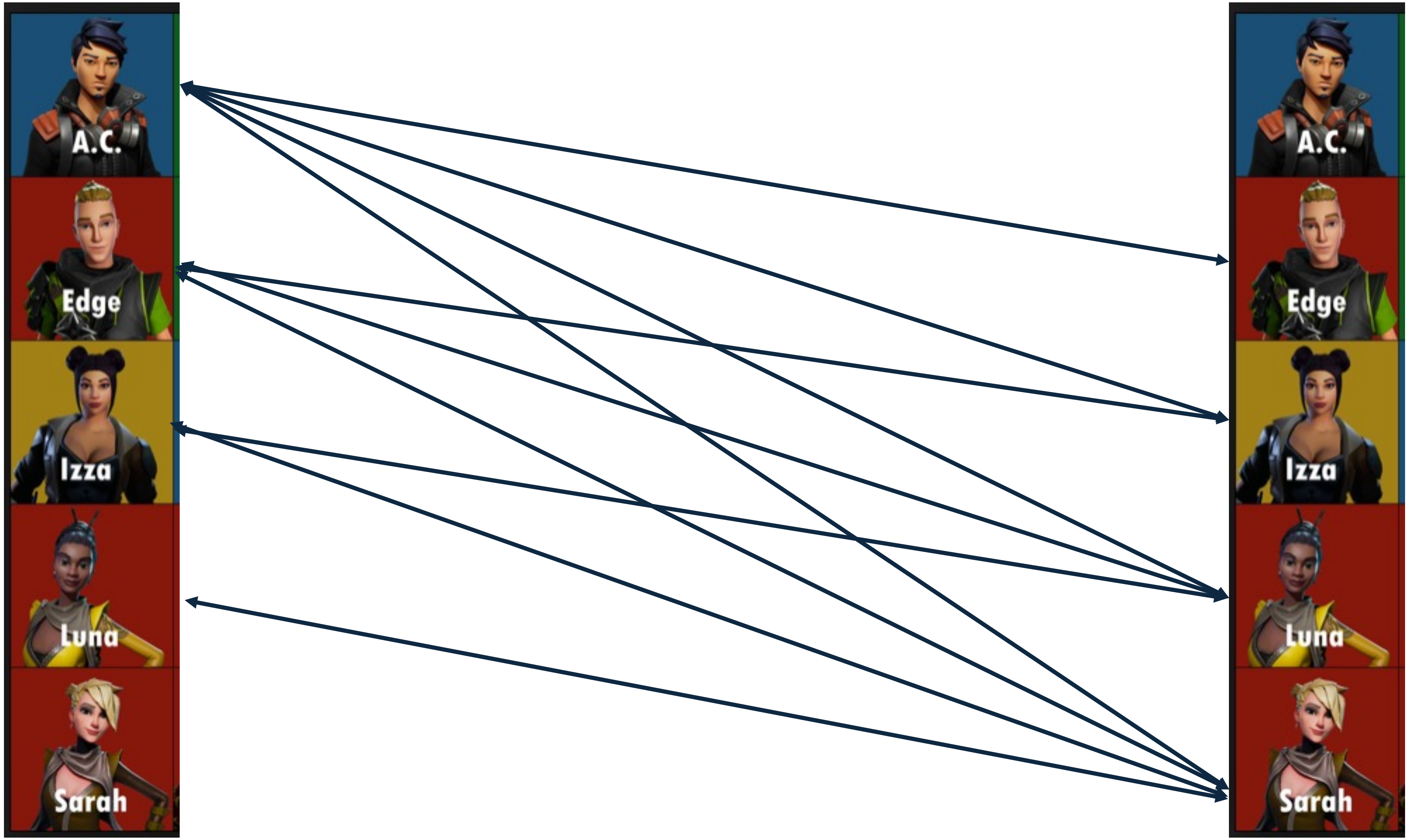
11 xor 65	22 xor 65	33 xor 65	44 xor 65	55 xor 66	9 xor 66	8 xor 66	7 xor 66	6 xor 66	5 xor 67	
74	87	96	109	118	75	74	69	68	70	CIPHERTEXT

Notice how the first A and the third B yield the same ciphertext
XOR (65, 11) == XOR (66, 8) = 74

Symmetric Key Challenges

- All the keys must be known by the sender and the receiver prior to secure communication.
- What if we have a cryptographic enterprise of 5 users?
- How many different keys would be in the enterprise to allow all 5 users to communicate with each other?

Key Distribution in Symmetric Key



N = number of users
K = number of keys
K = N(N-1) / 2

Symmetric Key Challenges

- This might work for 5 users, but what about a cryptographic system of 2,500 users?

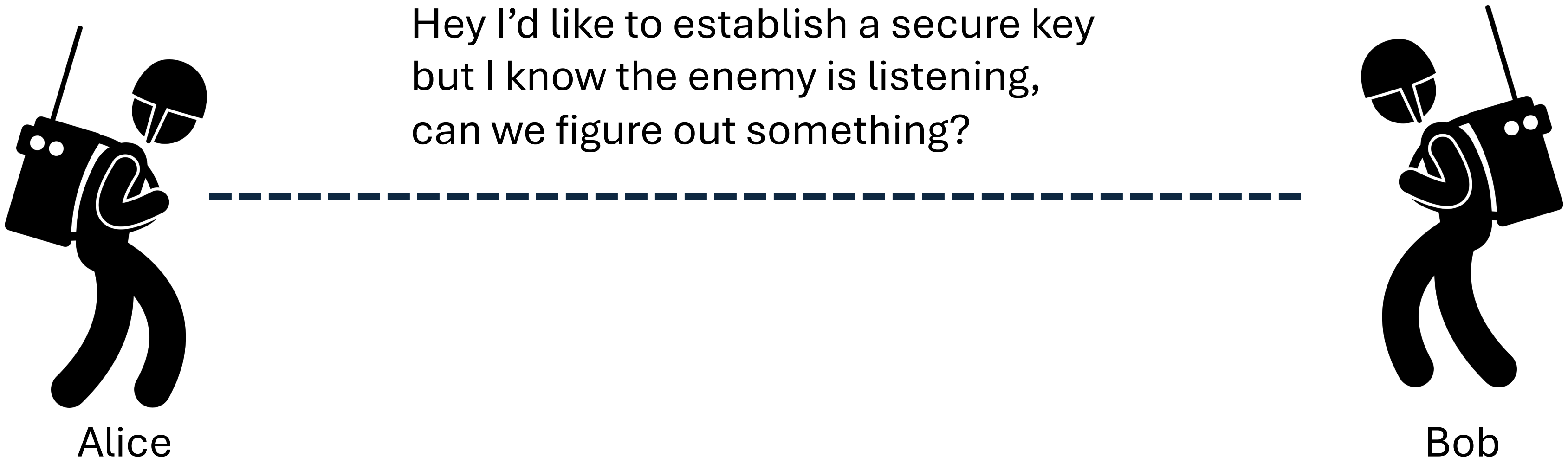
- **2,500 users would need to produce 2,500 * (2,500-1) / 10 = 624,750 keys**.

- And they'd all have to be shared prior to any secure communication.

Symmetric Key Challenges

- Storing and distributing 624,750 keys for 2,500 users presents some challenges.
- A centralized database of keys presents risks to users. What if the database is compromised or abused?
- Is there a better way to distribute 624,750 between 2,500 users?

Is there a better way?



Diffie-Helman Key Exchange

Alice

Bob

a = Alice secret

g = primitive root

p = prime

A = g^a mod p

b = Bob secret

g,p,A

B

B = g^b mod p

K = B^a mod p

K = A^b mod p

Lets consider two users Alice and Bob. They need to establish a Key (K) but know the enemy is listening to them.

1. Alice and Bon pick their own secret numbers (a,b).

2. Alice then picks a primiative root (g) and a prime number (p) and sends (g,p,g^a mod p) to Bob

3. Bob then sends g^bmod p to AC.

4. Both Alice and Bob calculate the same key K

Diffie Helman Key Exchange

Alice's Key = $B^a \bmod p = (g^b \bmod p)^a \bmod p$
Bob's Key = $A^b \bmod p = (g^a \bmod p)^b \bmod p$

Since $(g^b \bmod p)^a = (g^a \bmod p)^b$
Alice's Key == Bob's Key

Diffie-Helman Key Exchange

Alice

$a = 56789$

$g = 2$

$p = 104729$

$A = g^a \bmod p = 2^{56789} \bmod 104729 = 8836$

$K = B^a \bmod p = 31321^{56789} \bmod 104729 = 15300$

Bob

$b = 98765$

$B = g^b \bmod p = 2^{98765} \bmod 104729 = 31321$

$K = A^b \bmod p = 8836^{98765} \bmod 104729 = 15300$