Diffie Hellman and RSA

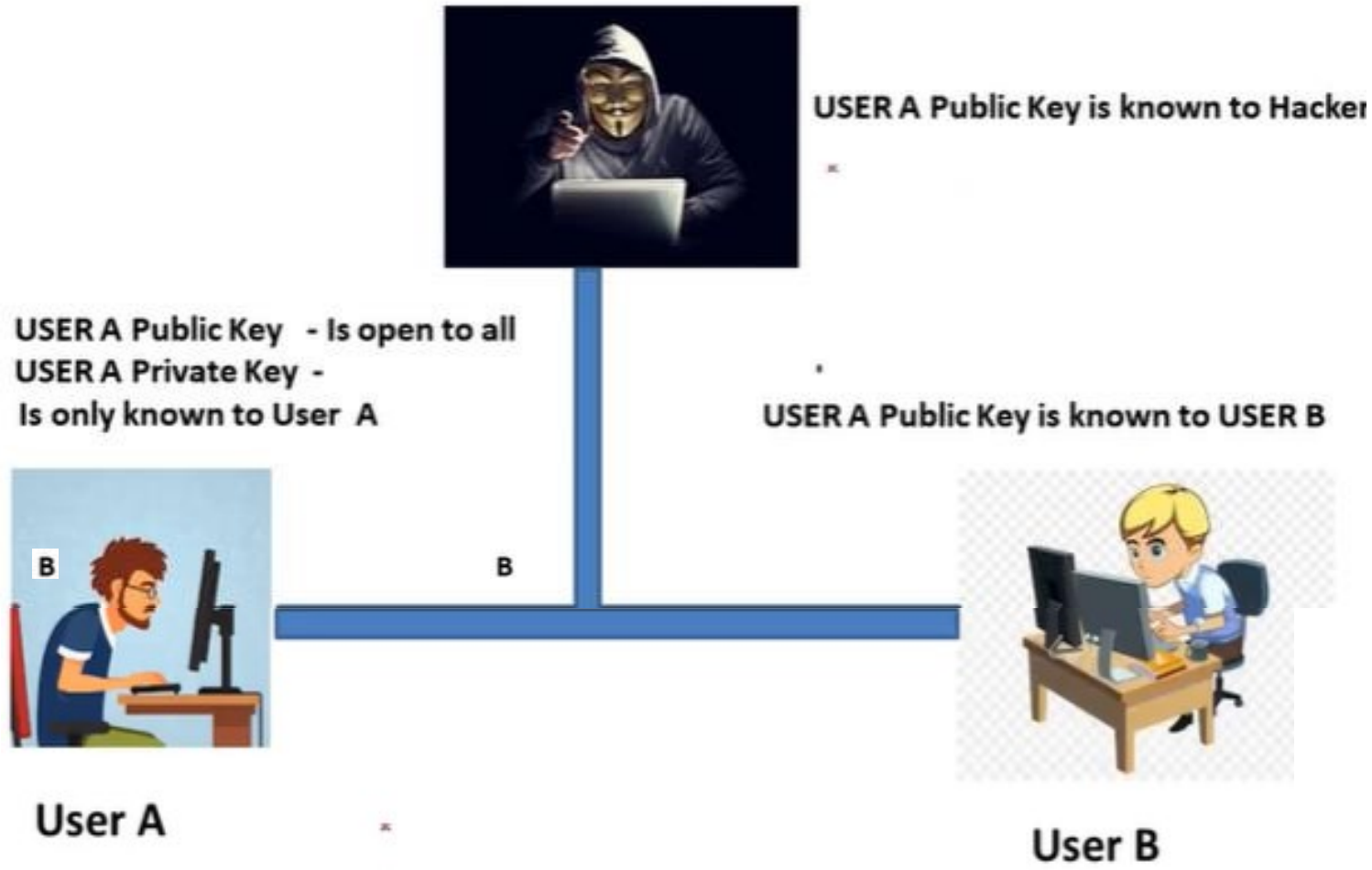
Public Key Cryptography

Public Key Cryptography

- Public – key cryptography or asymmetric cryptography is a cryptographic system that uses pairs of keys:
- Public Keys: that may be disseminated widely i.e., available publicly.
- Private Keys: that are known to the owners only.
- In such system, any person can encrypt a message using the receivers' public key, but that encrypted message can only be decrypted with the receiver’s private key.

Public Key Cryptography

- User A Private key is A
- User A Public Key is B
- User B use the Public Key of User A to encrypt message
- User A decrypt message Using his Private Key



Elagmal Cryptosystem

Elagmal Cryptosystem

- It is a public key encryption algorithm
- Proposed by Taher Elagmal in 1985
- It is based on Diffie-Hellman Key exchange
- It has three steps
    1. Generate Keys: generating public and private keys
    2. Encryption using public key
    3. Decryption using private key

Elagmal Cryptosystem

- Let there are two agents X and Y
- Agent A and B agreed on selecting generator = 2 and prime number = 13. In reality, the prime number is a large prime number.
- Generator g must be a primitive root of the prime number p, and Gcd(g, p) = 1
- The agent A select a secret value d, such that g < d < p-g
- Now calculate $e = g^d \bmod p$
- Public Key is [p, g, e]
- Private Key is [d]

Elagmal Cryptosystem

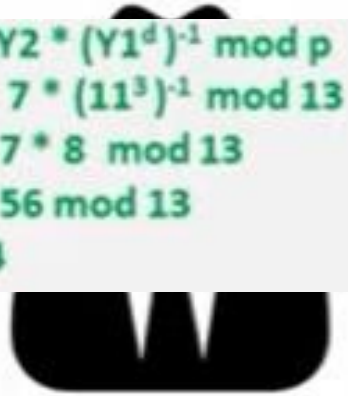
Prime Number
p = 13
Generator g= 2
Gcd(g,p)=1

Select d $2 \leq d \leq p-2$
d= 3
$e = g^d \bmod p$
$8 = 2^3 \bmod 13$
e=8

Public Key – p=13, g=2, e=8

Private key d = 3

Plain text = $Y2 * (Y1^d)^{-1} \bmod p$
Plaint text = $7 * (11^3)^{-1} \bmod 13$
Plaint text = $7 * 8 \bmod 13$
Plaint text = $56 \bmod 13$
Plain text = 4



Agent X

Agent X
Private Area

Problem

Enemy Scanning the Communication

Public Key – p=13, g=2, e=8

Y1 = 11

Y2 = 7

Public Area

Agent Y wants to Send
Message M= 4 to Agent X
M should be less than p

Select k a random integer (k=7)

$Y1 = g^k \bmod p$
$Y1 = 2^7 \bmod 13$
$Y1 = 11$

$Y2 = M * e^k \bmod p$
$Y2 = 4 * 8^7 \bmod 13$
$Y2 = 7$

Agent Y

Agent Y
Private Area

Elagmal Cryptosystem

- **What the enemy have to know to decrypt this cipher text is**

= 7 * (11^{d}) mode 13

He must know the value of d. If the values of p and Y1 are very large then it is almost impossible to compute it using discrete algorithms.

Elagmal Cryptosystem/ Generating Keys

- **Agent x chooses**

  I. A large prime p
  II. A primitive element g modulo p
  III. A (possibly random) integer d with $2 \leq d \leq p - 2$.
  IV. Computes $e = g^d \bmod p$
  V. Posts public key (p, g, e).
  VI. Private key is d.

Elagmal Cryptosystem/ Encryption

1. Agent Y encrypts a short message M (M < p) and sends it to Agent X like this:
2. Agent Y chooses a random integer k (which he keeps secret).
3. Agent Y computes **$Y1 = g^k \bmod p$** and **$Y2 = M * e^k \bmod p$**
4. Agent Y sends his encrypted message (Y1, Y2) to Agent X

Elagmal Cryptosystem/ Decryption

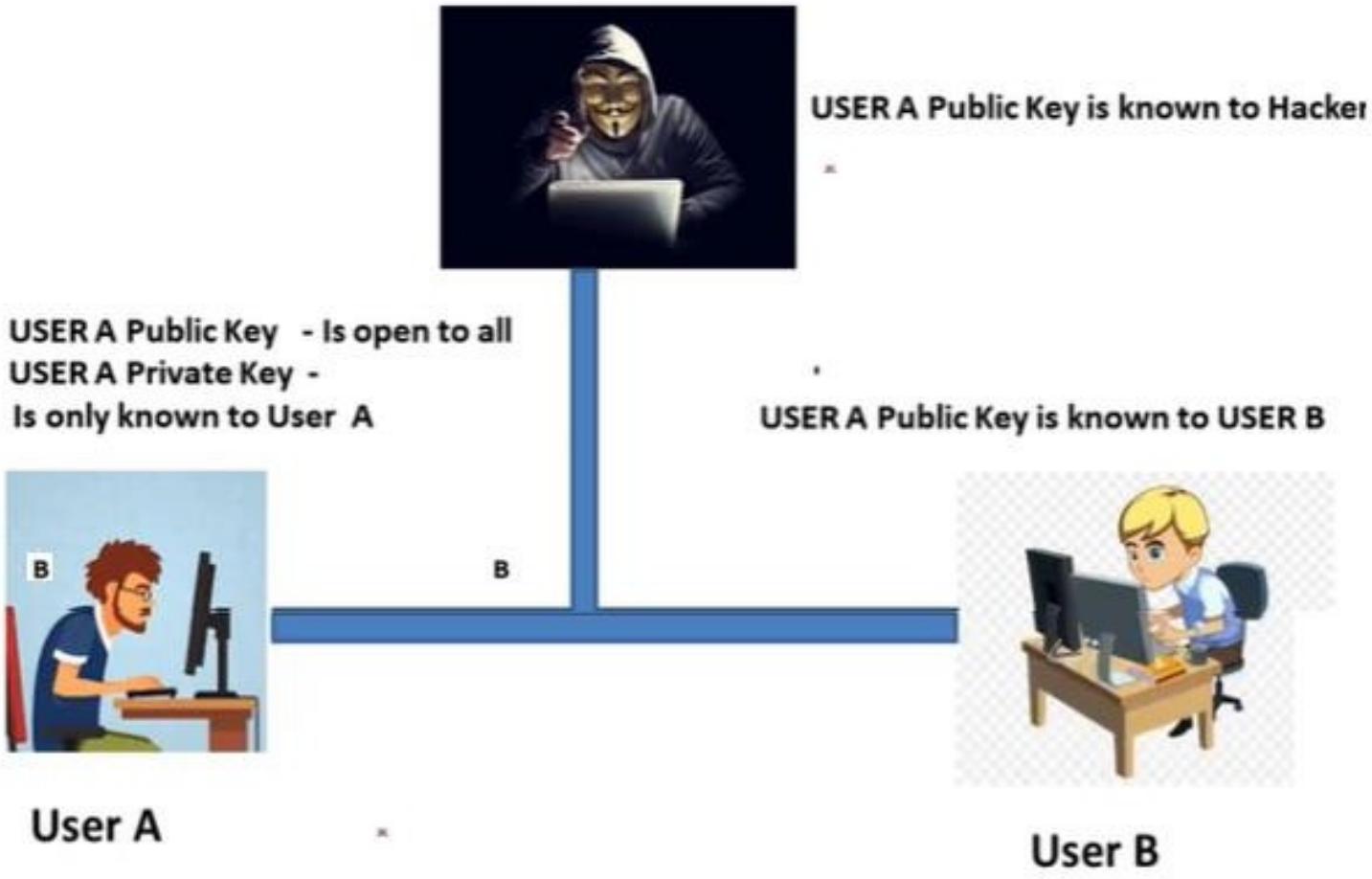
When Agent X receives the encrypted message (Y1, Y2), he decrypts (using the private key d) by computing

- **Plain text = $Y2 * (Y1^{d})^{-1} \bmod p$**

RSA Algorithm

RSA

- User A Private key is A
- User A Public Key is B
- User B use the Public Key of User A to encrypt message
- User A decrypt message Using his Private Key



RSA

- RSA (Rivest-Shamir-Adleman) is an algorithm used to encrypt and decrypt the messages.
- Introduced in 1977
- Asymmetric cryptographic algorithm
- It is also known as public key cryptography, because one of the key can be given to anyone.

Foundation of RSA

- What are the prime factors of 35?
  - Answer is 7 and 5.
  - What is the prime factor of RSA-250
  - RSA-250 =
    2140324650240744961264423072839333563008614715144755017797754920881418023447140136643345519095804679610992851872470914587687396261921557363047454770520805119056493106687691590019759405693457452230589325976697471681738069364894699871578494975937497937
  - Definitely it is very difficult.

Prime factors are:

p=6413528947707158027879019017057738908482501474294344720811685963202453234463023862359752668347708737661925585694639798853367

q=3337202759497815655622601060535511422794076034476755466678452098702384172921003708025744867329688187756571898625803693206271 1

RSA-250=p*q

Steps

- Generate Public Key and Private Key
- Encrypt using Public Key
- Decrypt using Private Key

Generate Public Key

- Select two prime numbers p and q
- Let n = p* q
- Select a number e such that e is $1<e<\phi(p)\ \phi(q)$ and gcd(e, $\phi(p)\ \phi(q)$) = 1
- The public key is {e, n}

Compute Euler's totient: $\varphi(n)=(p-1)\ (q-1)$

Generate Public Key

- Select two prime numbers p = 3 and q = 11
- Let n = p* q = 3 * 11 = 33
- Select a number e such that e is $1<e<\phi(p)\ \phi(q)$ and gcd(e, $\phi(p)\ \phi(q)$) = 1. It will be $\phi(p)\ \phi(q) = \phi(3)\ \phi(11) = (3-1)\ (11-1) = 20$ and e = 13 such that gcd(13, 20) =1
- The public key is {e, n} = {13, 33}

Generate Private Key

- Find a number d such that e * d mod $\phi(p)$ $\phi(q)$ = 1
  - d is the multiplicative inverse here
  - When you will have multiplicative inverse for a number in modular arithmetic then $e \cdot d \pmod{n} = 1$ is possible if gcd(e, n) = 1

Generate Private Key

- Find a number d such that e * d mod $\phi(p)$ $\phi(q)$ = 1

e * d mod $\phi(p)$ $\phi(q)$ = 1

13 * ? mod 20 = 1

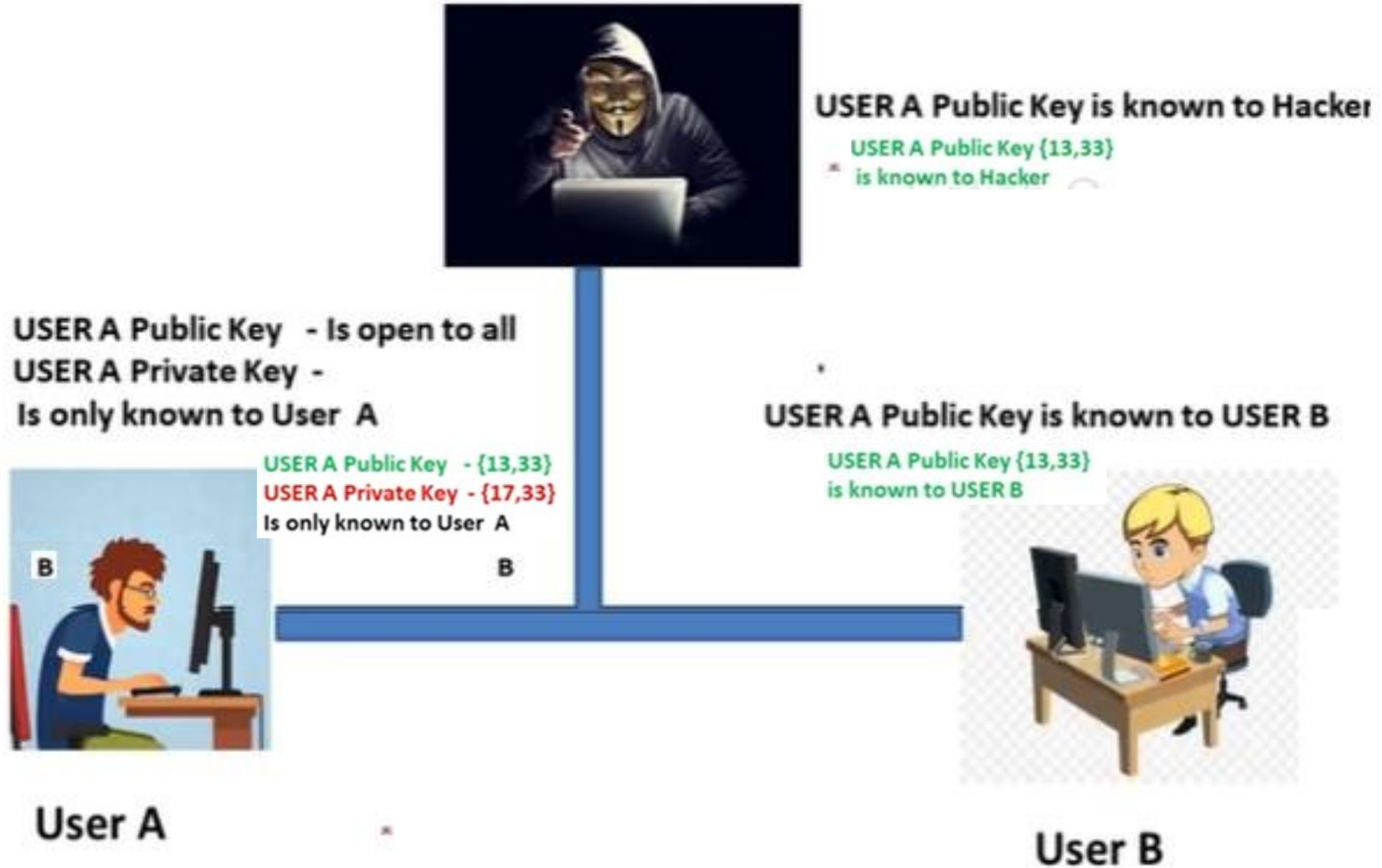
Here d = 17

13 * 17 mod 20 = 1

- The private Key is {17, 33}

Generate Private Key

- Public key known to everyone
- Private key known to User A Only.



Encryption

- User B knows the public key of A {e, n} = {13, 33}
- User B send number 4 to User A
- Plaint text p = 4 (p < n, it must be true)
- Encryption
- Cipher = $p^e \bmod n$

  $= 4^{13} \bmod 33 = 31$

Cipher text 31 will be sent to User A

Decryption

- User A receives ciphertext 31 from User B
- Private Key of User A {d, n} = {17, 33}
- Decryption
- Plaint text = $C^d \bmod n$

  $= 31^{17} \bmod 33 = 4$

Problems for Hacker

- Hacker do not know the private key of User A
- Hacker do not know the prime factors of n, because n is a very large number
- For example the prime factors of

RSA-250 =

2140324650240744961264423072839333563008614715144755017797754920881418023
4471401366433455190958046796109928518724709145876873962619215573630474547
7052080511905649310668769159001975940569345745223058932597669747168173806 9
364894699871578494975937497937

are

6413528947707158027879019017057738908482501474294344720811685963202453234 4
6302386235987526683477087376619255856946397988533 67

×

3337202759497815655622601060535511422794076034476755466678452098702384172 9
2100370802574486732968818775657189862580369320627 11

- visit

https://en.wikipedia.org/wiki/RSA_Factoring_Challenge

https://www.calculator.net/big-number-calculator.html

Resources

- DES https://www.scaler.com/topics/des-algorithm/
- AES https://www.simplilearn.com/tutorials/cryptography-tutorial/aes-encryption
- Diffie Hellman
- https://www.educba.com/diffie-hellman-key-exchange-algorithm/
- https://www.geeksforgeeks.org/implementation-diffie-hellman-algorithm/
- https://www.ques10.com/p/7533/explain-diffie-hellman-key-exchange-algorithm-wi-1/
- RSA
- https://www.geeksforgeeks.org/rsa-algorithm-cryptography/
- https://www.gatevidyalay.com/public-key-cryptography-rsa-algorithm/
- https://www.javatpoint.com/rsa-encryption-algorithm
- https://www.venafi.com/blog/how-diffie-hellman-key-exchange-different-rsa

Questions