# Computer Security
# Lecture 7

# RSA

Dr. Mohamed Loey
Lecturer, Faculty of Computers and Information
Benha University
Egypt

# Table of Contents

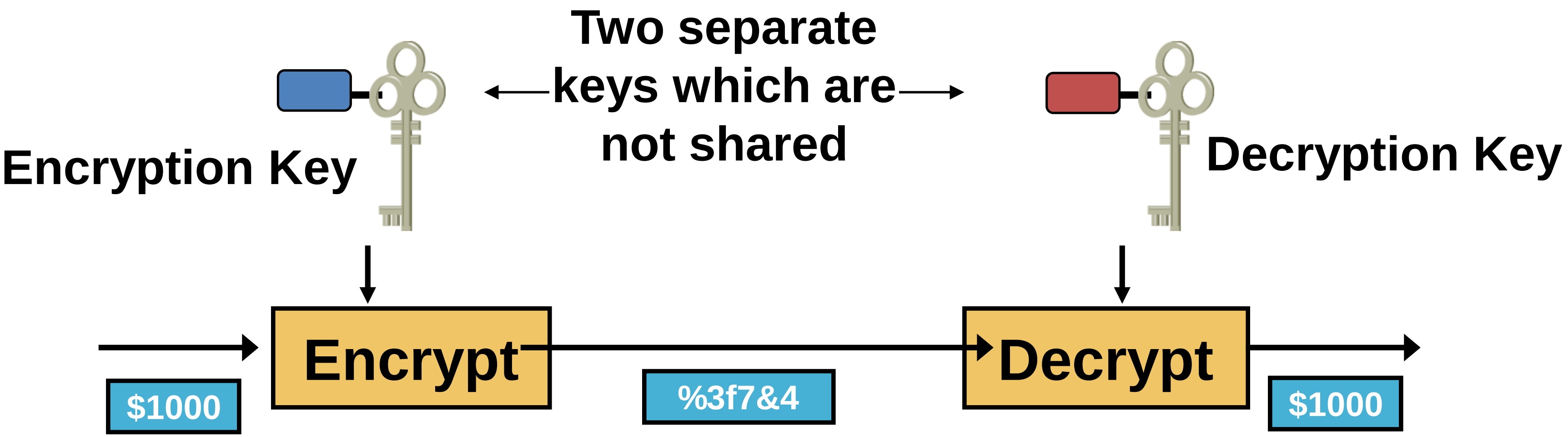
# Table of Contents

# RSA

- RSA is one of the first practical public-key cryptosystems and is widely used for secure data transmission.

- RSA is made of the initial letters of the surnames of [redacted] and Leonard Adleman, who first publicly described the algorithm in 1977.

- RSA is Asymmetric Encryption



# Table of Contents

# RSA Key generation

1) Choose two distinct prime numbers $\boldsymbol{p}$ and $\boldsymbol{q}$
2) Compute $\boldsymbol{n} = p * q$
3) Compute **φ(n)** = (p - 1) * (q - 1)
4) Choose **e** such that 1 < e < φ(n) and e and n are prime.
5) Compute a value for **d** such that (d * e) % φ(n) = 1

- ❑ Public key is **(e, n)**
- ❑ Private key is **(d, n)**

# RSA Key generation Example

- Choose $p = 3$ and $q = 11$
- Compute $n = p * q = 3 * 11 = 33$
- Compute $\varphi(n) = (p - 1) * (q - 1) = 2 * 10 = 20$
- Choose e such that $1 < e < \varphi(n)$ and e and n are prime. Let $e = 7$
- Compute a value for d such that $(d * e) \% \varphi(n) = 1$. One solution is $d = 3$ $[(3 * 7) \% 20 = 1]$
- Public key is $(e, n) => (7, 33)$
- Private key is $(d, n) => (3, 33)$

# Table of Contents

# RSA Encryption

- *m = plaintext*
- Public key is (e, n)
- *C= Ciphertext*

- $C = m^e \% n$

# RSA Encryption Example

- $m = 2$
- Public key is (e, n) => (7, 33)
- $C = 2^7 \% 33 = 29$

# Table of Contents

# RSA Decryption

- *C= Ciphertext*
- *m = plaintext*
- Private key is (d, n)

- $m = C^d \% n$

# RSA Decryption Example

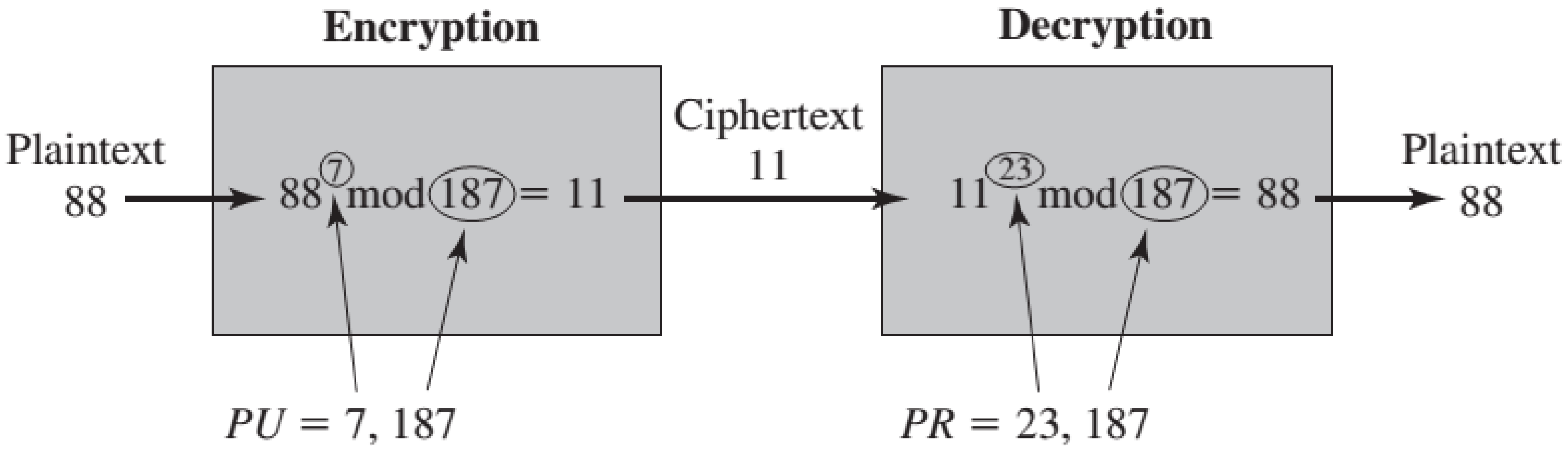
- *C= 29*
- Private key is (d, n) => (3, 33)

- $m = C^3 \% 33 = 2$

# RSA Another Example

- Select two prime numbers, $p = 17$ and $q = 11$.
- Calculate $n = pq = 17 * 11 = 187$.
- Calculate $\varphi(n) = (p - 1)(q - 1) = 16 * 10 = 160$.
- Select $e$ such that $e$ is relatively prime to $\varphi(n) = 160$ and less than $\varphi(n)$; we choose $e = 7$.
- Determine $d$ such that $d.e \equiv 1 \pmod{160}$ and $d < 160$. The correct value is $d = 23$, because $23 * 7 = 161 = (1 * 160) + 1$
- *Public Key* = {7, 187} and Private Key = {23, 187}

# RSA Another Example



# Table of Contents

# A Real World Example

- lets encrypt the message "attack at dawn“
- Convert the message into a numeric format. Each letter is represented by an ASCII character.
- "attack at dawn" becomes 1976620216402300889624482718775150

# A Real World Example

- p=

  121310724392112718973236715316124404284724276337014 10

  925634549312301964373042085619324197365322416866541 01

  70573613652141717117137979742993348710628298 03541

- q=

  120275242554787488859562207937345121287333878036820 75

  433653899983955179850988797899869146900809131611153 34

  681705083209602216014636634639181247098710541 5233

# A Real World Example

- **n=**1459067680075833232301869393490706352924018723753571643995818
  7101987343879900535893836957140267014980212181808629246742282815
  7022922076746906543401224889672472407926969987100581290103199317
  8587536637108623576565105078837142971156373427889114635351027120
  32765166518411726859837988672111837205085526346618740053

- **φ(n)**=14590676800758332323018693934907063529240187237535716439958
  1871019873438799005358938369571402670149802121818086292467422828
  1570229220767469065434012248896483138112322279966317301397778523
  6530154784827347887129722205858745715289160645926971811926897116
  3555070802643999529549644116811947516513938184296683521280

# A Real World Example

- **e= 65537**

- **d**=8948942500927444436822854592177309391966958606588425744549785445648767483962981839093494197326287961679797060891728367987549933157416111385408881327548811058824719307758252727843790650401568062342355006724004246666565423238350292221549362328947213886644581878912794612340780772570262664409103650237254513971 3

# A Real World Example

- ❑ **Encryption:** C= 1976620216402300889624482718775150 **^e % n**

- ❑ **C=**

  **3505211133867302669021242393705332851188076081157998162064280234668581062310985023594304908097338624111378404079470419397821537849976541308364643878474095230693253494519508018386157422522621887982723245391282059688644037753608246568175007441745915148540744586251102347223556082305349779151892882027225778778 6**

# A Real World Example

- **Decryption**: P= 35052111338673026690212423937053328511880760811579981620642802346685810623109850235943049080973386241113784040794704193978215378499765413083646438784740952306932534945195080183861574225226218879827232453912820596886440377536082465681750074417459151485407445862511023472235560823053497791518928820272257787786 ^d % n
- P= 1976620216402300889624482718775150 (which is our plaintext "attack at dawn")

# Table of Contents

# RSA Security

- Two approaches to attacking RSA:
  - brute force key search (infeasible given size of numbers)
  - mathematical attacks (based on difficulty of computing ø(N), by factoring modulus N)

# Contact Me

facebook.com/mloey

mohamedloey@gmail.com

twitter.com/mloey

linkedin.com/in/mloey

mloey@fci.bu.edu.eg

mloey.github.io

# THANKS FOR YOUR TIME