Security of Public Key Schemes

- Keys used are very large (>512bits)
  - like private key schemes brute force **exhaustive search** attack is always theoretically possible
- Security relies on a large enough difference in difficulty between easy (en/decrypt) and hard (cryptanalyze) problems
  - more generally the hard problem is known, it’s just made too hard to do in practice
- Requires the use of very large numbers, hence is slow compared to private/symmetric key schemes

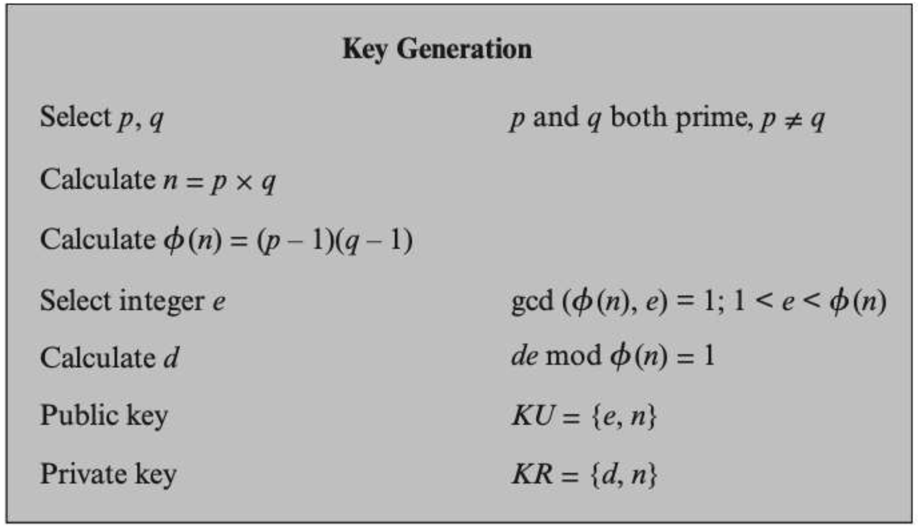
Public-Key Cryptography Algorithm
(RSA)

RSA Public-key encryption

- by Rivest, Shamir & Adleman of MIT in 1977
- currently the “work horse” of Internet security
  - most public key infrastructure (PKI) products
  - SSL/TLS: certificates and key-exchange
  - secure e-mail: PGP, Outlook, ….
- based on exponentiation in a finite (Galois) field over integers modulo a prime
  - exponentiation takes $O((\log n)^3)$ operations (easy)
- security due to cost of factoring large integer numbers
  - factorization takes $O(e^{\log n \log \log n})$ operations (hard)
- uses large integers (eg. 1024 bits)

RSA key setup

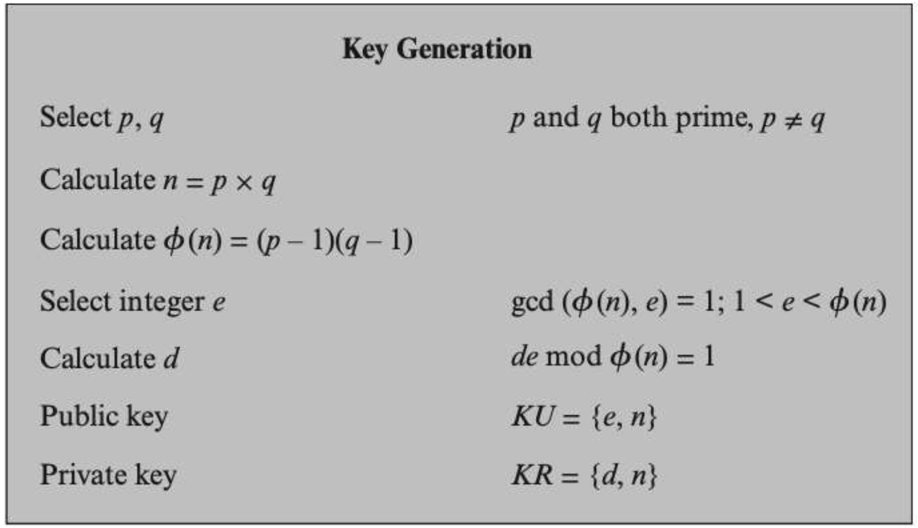
- each user generates a public/private key pair by:
  - selecting two large primes at random - p, q
  - computing their system modulus $n=p\cdot q$
    - note $\varnothing(n)=(p-1)(q-1)$
  - selecting at random the encryption key e
    - where $1<e<\varnothing(n)$, $\gcd(e,\varnothing(n))=1$
  - solve following equation to find decryption key d
    - $ed=1 \bmod \varnothing(n)$
  - publish their public encryption key: pk={e,n}
  - keep secret private decryption key: sk={d,p,q}



Key Generation

Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$de \bmod \phi(n) = 1$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

RSA example



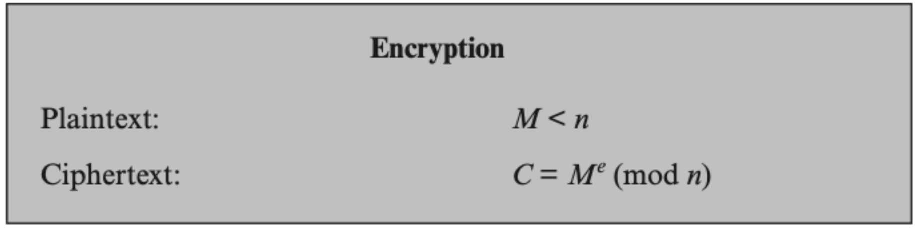
Key Generation

Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$de \bmod \phi(n) = 1$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

1. Select primes: $p=17$ & $q=11$
2. Compute $n = pq = 17\times11=187$
3. Compute $\varnothing(n)=(p-1)(q-1)=16\times10=160$
4. Select e : $\gcd(e,160)=1$; choose $e=7$
5. Determine d: $de=1 \bmod 160$ and $d < 160$ Value is $d=23$ since $23\times7=161= 10\times160+1$
6. Publish public key $pk=\{7,187\}$
7. Keep secret private key $sk=\{23,17,11\}$

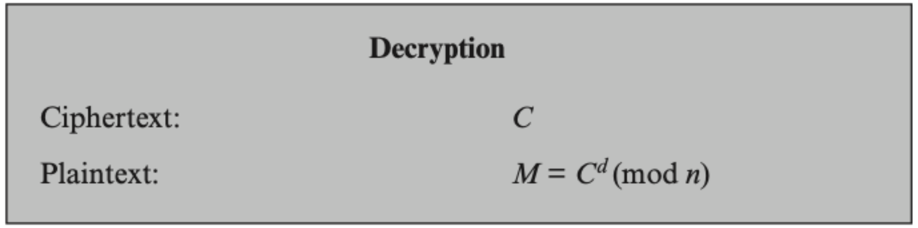
RSA use

- to encrypt a message M the sender:
  - obtains **public key** of recipient `pk={e,n}`
  - computes: $C=M^e \bmod n$, where $0 \leq M < n$
- to decrypt the ciphertext C the owner:
  - uses their private key `sk={d,p,q}`
  - computes: $M=C^d \bmod n$
- note that the message M must be smaller than the modulus n (block if needed)



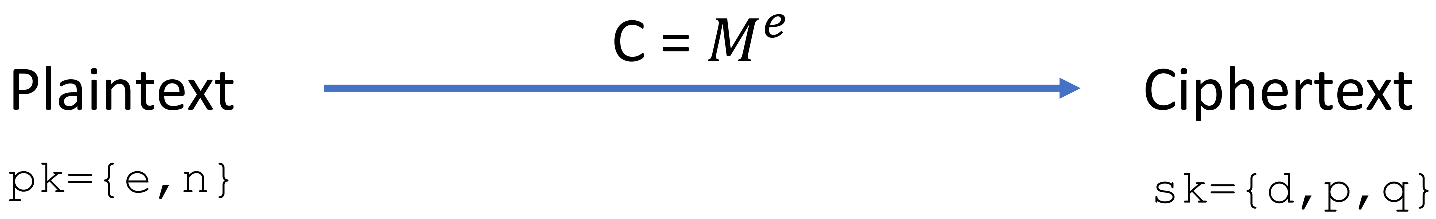
Encryption

Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod{n}$



Decryption

Ciphertext:	C
Plaintext:	$M = C^d \pmod{n}$



C = M^e
Plaintext
Ciphertext
pk={e,n}
sk={d,p,q}