



### Problem: Key Exchange

To communicate securely, users must meet before sending/receiving

Sender Encrypt w/ $K_{sr}$  → Cipher Text  $E_{sr}(T)$  → Receiver Decrypt w/ $K_{sr}$

Clear Text  $T$  → Clear Text  $D_{sr}(E_{sr}(T)) = T$

This doesn't work for eCommerce

### Revise Encryption Setup

Public Key Encryption is based on publishing the key

Sender uses public key to encrypt

Sender Encrypt w/ $K_p$  → Cipher Text  $E_p(T)$  → Receiver Decrypt w/ $K_r$

Clear Text  $T$  → Clear Text  $T = D_r(E_p(T))$

### Public Key Encryption

Using the public key, encrypt message

- Divide  $T$ , the clear text bytes, into blocks
- Treat each block as a number
- Cube number (raise to 3 pwr), divide by key
- Send the remainder for each block

$T$ : 100010110101111010000010010000101101010100101

2229	3714	1069	165
$2229^3$	$3714^3$	$1069^3$	$165^3$
11074654989/ $K_p$	51230158344/ $K_p$	1221611509/ $K_p$	4492125/ $K_p$
25	24	19	0

### Public Key Cryptography

Does PKC work? Can't it be cracked?

- Recall definition of divide:  $a = b \cdot c + d$

Quotient      Remainder

- For example,  $50/6$  implies  $50 = 6 \cdot 8 + 2$
- The encryption process is a division:  $T^3 = K_p \cdot c + d$

so sending  $c$  &  $d$  determines clear text  $T$

But we only send  $d!$

### RSA Encryption

Rivest, Adelman and Shamir invented a PKC scheme called RSA

- The secret is to pick the key,  $K_p$ , right
- Pick two prime numbers -- numbers divisible only by themselves and 1 -- that are 2 greater than a multiple of 3 ... weird!
- Examples are 5, 11, 17, 23, 29, ...
- $K_p = p \cdot q$  so that it is 129 digits

Follow procedure given, send remainders

### How To Recover Message

Compute  $s = 1/3(2(p-1)(q-1)+1)$  then compute  $C^s = K_p \cdot c + T$

\* That is ...

The remainders ( $C$ ) raised to  $s$  power equal  $K_p$  times some (quotient)  $c$  no one cares about plus the original clear text number!

- So, raise the remainders to  $s$ , divide by  $K_p$  and PRESTO! the new remainder is the answer

For  $p=17$  and  $q=23$ ,  $pq=391$  and  $s=235$



## What Makes RSA Work?

Though the numbers get huge,  
computer can handle them quickly

- These codes are strong because breaking them needs  $s$ , which needs  $p, q$ , which means factoring  $K$
- Factoring is computationally tough -- best methods are only somewhat better than grammar school, "try all small primes"
- Picking 129 digit key, means no computer can factor it ... so the code is unbreakable



## RSA Challenge

After inventing their scheme (1977),  
RSA challenged people to break it

- Their first key was broken in 1994 using 1000 computers over 8 months
- Their secret message: THE MAGIC WORDS ARE SQUEAMISH OSSIFRAGE  
*Doomed? No. There are many other 129 digit keys, or if people get nervous make 200 digit keys or more ... breaking gets harder very fast; encrypt/decrypt doesn't*



## Is Strong Encryption Smart

Should we allow people to use strong  
encryption? Or should only  
breakable codes be legal?

- It hampers law enforcement and security
- Most criminals reveal plans in other ways
- PKC exists and is known, so build in escape
  - Trap door
  - Key Escrow
- But are these schemes really secure?