Key

Why is there, at all, such a thing?
Why is it not 42 ?
Why is the smallest public key not 35 ?
Or, for that matter, 6 ?

## Shachar Shemesh <br> Lingnu Open Source Consulting

(1)(1)(1)

## Introducing the Players

## s Alice and Bob

B Want to communicate.
B In this lecture, we'll assume Alice wants to send a message to Bob without anyone listening.
© Eve
B The attacker
© In our case, we'll assume she wants to intercept and understand the message sent from Alice to Bob.
$\Delta$ Alice and Bob have a shared secret.
$\Delta$ Usually a totally random set of bits.
B This secret is called "the key".
s The key has to be shared via a secure channel.
B If Eve learns of the key, she will be able to decipher the message.
Q Alice's and Bob's keys are the same.

## Asymmetric Encryption

$\Delta$ Alice and Bob are communicating over insecure line.
© Eve can listen in at will.
$\Delta$ Alice describes the algorithm to Bob.
$\Delta$ Bob generates a key and tells it to Alice.
$\Delta$ Alice encrypts the message and sends it to Bob.
$\Delta$ Eve hears the algorithm, the key and the encrypted message, and yet cannot know what Alice's message was.

## First Algorithm of Family

$\Delta$ Bob encrypts $2^{28}$ blocks with known P and unique I.
Q Bob sends all of the encrypted data, as well as all but 28 bits of each key, to Alice.
B Alice chooses an encrypted block at random, and invests the CPU required to brute force the missing bits from the key.
Q Alice sends the corresponding I to Bob.
© Bob now knows which block Alice picked, and they have a shared secret.
$\Delta$ Eve has to brute force all blocks in order to find the right one.

## Merkle's Puzzles Analysis

$\Delta$ Bob has to perform $2^{28}$ encryptions in order to encrypt all blocks.
© Alice has to perform around $2^{28}\left(2^{27}\right.$ on average) to brute force the missing 28 bits of the key.
$\Delta$ Eve has to brute force ( $2^{27}$ on average) an average of half the blocks ( $2^{27}$ ) in order to find the key $=2^{54}$, can get as bad as $2^{56}$.
$\Delta$ All in all, both Alice and Bob invest the square root of what Eve has to invest in order to agree on a key.

## Background - What is RSA

8 Named after its inventors:
B Ron Rivest
S Adi Shamir
© Leonard Adleman
© Public Key encryption
Q Public key used for encryption
Q Private key used for decryption
$\Delta$ No easy way to get from public to private key

## Mathematically Based Encryption

© Unlike the Merkle's Puzzles above, Public Key encryption is based on mathematical principles.
\& Two popular mathematical principles:
\& Discrete root - the basis for RSA
© Discrete log - the basis for DSA, ElGamal and Diffie Hellman.
$\Delta$ Public key keys cannot be any number.
$\Delta$ As the algorithm is mathematical, the keys have to keep some mathematical properties.
B Typically, 64 bits is an ok size for a symmetric key. An RSA key should be no less than 512 bits, 1024 as preference.

# Open Source Consulting <br> <br> RSA Mathematical Principles 

 <br> <br> RSA Mathematical Principles}

Modern algebra, here we come...
© A quick reminder:
$\mathbf{Q} \mathbb{N}$ - Natural numbers (1, 2, 3 etc.)
$\Delta \mathbb{Z}$ - Whole numbers ( $\mathbb{N}+$ negatives and zero)
$\Delta \mathbb{Q}$ - Rational numbers. Any number that can be denoted as a division of two numbers from $\mathbb{Z}$
$\Delta \mathbb{R}$ - Real numbers.
$\Delta \mathbb{C}$ - Complex numbers.
B All of the above groups are infinite in size
$\Delta \mathbb{N}, \mathbb{Z}$ and $\mathbb{Q}$ are $\aleph_{0}$
$\Delta \mathbb{R}$ and $\mathbb{C}$ are

## Modern Algebra - Modulus Group

© What happens if we limit ourselves to a finite sized group?
\& We'll call it "all the whole numbers smaller than $n$ ", and mark it with $\mathbb{Z}_{n}$.
$\Delta$ E.g. $-\mathbb{Z}_{5}$ will be the group $\{0,1,2,3,4\}$
$\Delta$ Arithmetics "+" and " $\times$ " are defined as with $\mathbb{Z}$, taking the reminder (modulus) from $n$.
$\Delta$ The operation is marked accordingly with $\bmod n "$.
\$ We use the congruent ( $=$ ) sign instead of equal ( $=$ )
$3+4=7=2 \bmod 5$
$3 \times 3=9 \equiv 4 \bmod 5$

## Inversing the + Operator

$\Delta$ Inversing the "+" operator is no more than applying the "+" semantics to "-".
© Simply perform the same operation you would on $\mathbb{Z}$, and then take the positive modulus $n$ of the result.
$\Delta 1-4=2 \bmod 5$
81-4 $=3 \bmod 6$
81-4=4 mod 7

## Greatest Common Divisor

$\Delta \operatorname{gcd}(a, b)$ is the biggest number that divides both $a$ and $b$.
$\Delta \operatorname{gcd}(12,4)=4$
$\Delta \operatorname{gcd}(15,12)=3$
$\Delta \operatorname{gcd}(15,8)=1$
© Numbers that have a gcd of 1 are called "coprime".
Q It means they have no common prime factors.
© A close relative - Least Common Multiple
$\Delta$ For two numbers, $\operatorname{lcm}(a, b)=a \cdot b / \operatorname{cdd}(a, b)$
$\Delta$ To compute $\operatorname{gcd}(a, b)$ for $a>b$
$\Delta$ Compute $a=c_{1} \cdot b+r_{1}$, where $r_{1}<b$ and all numbers are whole.
$\Delta$ Compute $b=c_{2} \cdot r_{1}+r_{2}$
B Repeat until $r_{i}$ is zero (in other words, until $r_{i-1}$ cleanly divides $\left.r_{i-2}\right)$.
© When that happens, $r_{i-1}$ is the gcd.

# Example - gcd $(21,102)$ 

$102=4 \cdot 21+18$
$21=1 \cdot 18+3$
$18=6 \cdot 3+0$
$\operatorname{gcd}(21,102)=3$

## A Useful Trick

## Euclid's Extended Algorithm:

Any two numbers $a$ and $b$ will have whole numbers $p$ and $r$ such that $p \cdot a+r \cdot b=\operatorname{gcd}(a, b)$

## Example - gcd $(21,102)$

$102=4 \cdot 21+18$
$21=1 \cdot 18+3$
$3=21-1 \cdot 18$
$18=6 \cdot 3+0$
$\operatorname{gcd}(21,102)=3$
$18=102-4 \cdot 21$

.

$$
\begin{gathered}
3=21-1 \cdot 18 \\
=21-1 \cdot(102-4 \cdot 21) \\
=5 \cdot 21-1 \cdot 102
\end{gathered}
$$

## Inversing the $\times$ Operator

$\Delta$ We seek " $b$ " such that $b \times a \equiv c$ mod $m$.
s In particular, we would like to find the solution for $b \times a \equiv 1 \bmod m$. We call $a$ "the inverse of $b "$, or $b^{-1}$.
© Example: $5^{-1} \equiv 3 \bmod 7$

$$
\Delta 3 \times 5=15 \equiv 1 \bmod 7
$$

$\Delta$ Let's try to find $2^{-1}$ over $\mathbb{Z}_{4}$ :
$2 \times 0 \equiv 0 \bmod 4,2 \times 1 \equiv 2 \bmod 4,2 \times 2 \equiv 0 \bmod 4,2 \times 3 \equiv 2 \bmod 4$
Q On the other hand:
S $3 \times 3=9 \equiv 1 \mathrm{mod} 4$
$\Delta 3$ is its own inverse mod 4.
s In order for $a$ to have an inverse in $\mathbb{Z}_{n}$, it must be coprime to $n$. In other words, $a^{-1}$ exists IFF $\operatorname{gcd}(a, n)=1$
$\Delta$ If $a$ is coprime to $n, a$ is also called a "generator" of $\mathbb{Z}_{n}$.
$\Delta a \times i \bmod n$ will generate all members of $\mathbb{Z}_{n}$ when $i$ goes from 0 to $n-1$.

## How To Find an Inverse

$\Delta$ To find $a^{-1} \bmod n$
S Perform Euclid's extended algorithm to calculate $\operatorname{gcd}(a, n)$
$\Delta$ If the result is not 1 , there is no inverse.
$\Delta$ If the result is 1 , figure out $p$ and $r$ as before.
$\Delta$ We now have $p \cdot a+r \cdot n=1$
$\Delta$ Stated over $\mathbb{Z}_{n}$, this turns into $p \cdot a+r \cdot n \equiv 1 \bmod n$.
$\Delta$ However $n \equiv 0 \bmod n$ for every $n$.
$\Delta$ We are left with $p \cdot a+r \cdot 0 \equiv 1 \bmod n$, which can also be written as $p \cdot a \equiv 1 \bmod n$.
$\Delta$ It follows that $p$ is $a$ 's inverse.

## Example - Calculate $9^{-1} \bmod 16$

$$
\begin{aligned}
& 16=1 \cdot 9+7 \\
& 9=1 \cdot 7+2 \\
& 7=3 \cdot 2+1 \\
& 2=2 \cdot 1+0 \\
& \operatorname{gcd}(16,9)=1
\end{aligned}
$$

$$
\begin{gathered}
7=16-1 \cdot 9 \\
2=9-1 \cdot 7 \\
1=7-3 \cdot 2=7-3 \cdot(9-1 \cdot 7) \\
=-3 \cdot 9+4 \cdot 7 \\
=-3 \cdot 9+4 \cdot(16-1 \cdot 9) \\
1=4 \cdot 16-7 \cdot 9 \\
9-1=-7 \equiv 9 \\
\text { mod } 16
\end{gathered}
$$

$\Delta$ We denote by the function $\varphi(n)$ (Greek letter "phi") the number of numbers smaller than $n$ that are coprime to it.
$\Delta \varphi$ is easy to calculate in certain cases:
$\Delta$ For any prime $p$ we have $\varphi(p)=p-1$
$\Delta$ If $a$ and $b$ are coprime, then $\varphi(a \cdot b)=\varphi(a) \cdot \varphi(b)$

## Fermat's Theorem (Not That One)

$\Delta$ For every prime integer $p$ and any $a$, we can say that $a^{p} \equiv a \bmod p$.
$\Delta$ Put another way, $a^{p-1} \equiv 1 \bmod p$
s Often called "Fermat's little theorem"
© We can also apply Euler's more general theorem:
$\Delta a^{\varphi(n)} \equiv 1 \bmod n$
$\Delta$ Applies for any $n$, prime or otherwise.
$\Delta a$ must be coprime to $n$.

## Chinese Reminder Theorem

$\Delta$ It is possible to find $x$ such that:
$x \equiv a_{1} \bmod n_{1}$
$x \equiv a_{2} \bmod n_{2}$
$x \equiv a_{i} \bmod n_{i}$
for a known set of $a_{i}$ and $n_{i}$.
\& The constants have to conform to a certain consistency rule.
$\Delta$ If all $n$ s are coprime in pairs, this rule is guaranteed.
$\Delta x$ will repeat every $\operatorname{lcm}\left(n_{1}, n_{2}, \ldots n_{i}\right)$
$\Delta$ Reminder - if all $n$ s are coprime in pairs, $\operatorname{lcm}\left(n_{1}, n_{2}, \ldots n_{i}\right)$ is simply $n_{1} \times n_{2} \times . . \times n_{i}$

And you thought this moment will never come....

## Encryption

$\Delta$ An RSA public key is composed of two numbers:
© Encryption exponent. We'll use " $e$ ".
© The actual public key. We'll call it " $n$ ".
s To encrypt the message " $m$ " into the encrypted form $M$, perform the following simple operation: $M=m^{e} \bmod n$
s When performing the power operation, actual performance greatly depends on the number of " 1 " bits in $e$.
© Originally used to use $e=3$.
B Today we usually use $e=2^{16}+1=65,537$

## Decryption

$\Delta$ In order to decrypt, we need to reverse the exponent used for encryption.
\& We know, from Fermat's and Euler's theorems that: $m^{\varphi(n)+1} \equiv m \bmod n$
$\Delta$ We have $M=m^{e} \bmod n$
$\Delta$ We need to find $d \equiv e^{-1} \bmod \varphi(n)$
$\Delta$ Decryption is merely:
$m \equiv M^{d} \bmod n$

## Selecting the Keys

$\Delta$ When selecting the public key $n$ we make sure that this will be possible.
$\Delta$ For one thing, we need to make sure that e and $\varphi(\mathrm{n})$ are coprime.
$\Delta$ In order to generate the keys we select two prime numbers. We'll call them $p$ and $q$.
s $n=p \times q$
$\Delta e=3$ (or 65,537 , as the case may be)
$\Delta \varphi(n)=(p-1)(q-1)$
$\Delta d=e^{-1}$ is calculated using Euclid's extended algorithm.
$\Delta$ First attempt - smallest primes. $p=2, q=3, n=6$.
$\Delta$ Problem - cannot encrypt. $\varphi(6)=(2-1)(3-1)=2.3^{-1} \equiv 1 \bmod 2$. In other words, to decrypt you need to raise by the power of " 1 ". In yet other words, $e$ does not encrypt. Each $m$ is mapped to itself.
$\Delta$ Second attempt - keep the primes bigger than e.p=5, $q=7 . n=35$
$\Delta$ Problem $-\varphi(35)=(5-1)(7-1)=24 . \operatorname{gcd}(e, \varphi(n))=\operatorname{gcd}(3,24)=3 \cdot e^{-1}$ doesn't exist.
$\Delta$ Must keep $\operatorname{gcd}(e, \varphi(n))$ by keeping $e$ and $p-1$ and $e$ and $q-1$ coprime.
$\Delta 5$ was ok as private key part $-\operatorname{gcd}(3,4)=1$. Next prime is 11 .

$$
\begin{gathered}
p=5 \\
q=11 \\
n=55 \\
e=3 \\
\varphi(n)=(5-1)(11-1)=40 \\
d=27
\end{gathered}
$$

## Example

Using a public key of 55 and an $e$ of 3 we encrypted a message $m^{3} \bmod 55$ and got $M=3$.

What was the original message?

## A Little Performance Trick

$\Delta$ When performing decryption, $p$ and $q$ are often known.
s Standard decryption method:
$m \equiv M^{d} \bmod n$
s Quicker decryption method:
$\Delta m_{1} \equiv M^{d} \bmod p$
$\Delta m_{2} \equiv M^{d} \bmod q$
$\Delta$ Use the Chinese reminder theorem to calculate $m \bmod n$

## Found the Minimal RSA Key

$$
\begin{gathered}
p=5 \\
q=11 \\
n=55 \\
e=3 \\
\varphi(n)=(5-1)(11-1)=40 \\
d=27 \\
d_{p} \equiv d \bmod p-1=27 \bmod 4=3 \\
d_{q} \equiv d \bmod q-1=27 \bmod 10=7
\end{gathered}
$$

## Example Decryption

$$
\begin{gathered}
M=3 \\
n=55 \\
m_{p} \equiv 3^{3} \bmod 5=2 \\
m_{q} \equiv 3^{7} \bmod 11=9 \\
m \equiv 42 \bmod 55
\end{gathered}
$$

## Encrypting Multiples of $p$ or $q$

$\Delta$ Euler's theorem only applies to numbers coprime to $n$.
Q We are not, at all, sure that we can decrypt such a message!
$\Delta$ Let's assume $m^{e}$ is an encrypted message, and that $m$ is a multiple of $p$.
$\Delta m \equiv 0 \bmod p$, therefor $m^{e} \equiv 0 \bmod p$
$\Delta$ We know that $d \equiv e^{-1} \bmod \varphi(n)$, which means $d \equiv e^{-1} \bmod q-1$.
$\Delta$ So we know that raising to the power of $d$ will do nothing $\bmod p$ (zero is unaffected), and will decrypt $\bmod q$ (due to Fermat's little theorem).
© Hence, these message will decrypt as well.

## RSA Security

$\Delta$ In order to decrypt Alice's messages, Eve needs to figure out $d$.
$\Delta$ No (known) efficient method of obtaining $d$ other than calculating $e^{-1} \bmod \varphi(n)$
$\Delta$ No (known) efficient method of calculating $e^{-1} \bmod \varphi(n)$ without knowing $\varphi(n)$.
$\Delta$ No (known) efficient method of calculating $\varphi(n)$ without knowing $p$ and $q$ ( $n$ 's factors).
$\Delta$ No (known) efficient method of factorizing $n$.
© No (known) method for breaking RSA.

Q Complexity of operations:
$\Delta \mathrm{O}(\operatorname{gcd}(a, b))=\log (\min (a, b))$ division operations.
© Some pretty effective probability algorithms for finding prime numbers.
© No efficient algorithm for factorizing a number.
© Eve needs to work non-polynomially harder than Alice and Bob in order to attack their keys.

Decrypting Messages Without $d$

## The Attack Scenario

© Alice has to send the same message to Bob, Charlie and Debbie.

B Each provided Alice with their respective public key.
© Unsurprisingly, they all use the same $e$ of 3 .
$\Delta$ Alice computes $m^{3} \equiv M_{\mathrm{b}} \bmod n_{\mathrm{b}}, m^{3} \equiv M_{\mathrm{c}} \bmod n_{\mathrm{c}}, m^{3} \equiv M_{\mathrm{d}}$ $\bmod n_{d}$.
$\Delta$ If Eve knows that $M_{\mathrm{b}}, M_{\mathrm{c}}$ and $M_{\mathrm{d}}$ were generated from the same $m$, she can obtain $m$ without knowing any of the required $d$ s.

## Attack Method

s Eve knows:
$m^{3}=M_{\mathrm{b}} \bmod n_{\mathrm{b}}$
$m^{3}=M_{\mathrm{c}} \bmod n_{\mathrm{c}}$
$m^{3} \equiv M_{\mathrm{d}} \bmod n_{\mathrm{d}}$
$\Delta$ Eve uses the Chinese reminder theorem to calculate $m^{3}$.
$\Delta$ Eve takes the regular $3^{\text {rd }}$ root of $m^{3}$.
$\Delta$ Eve knows $m$.
$\Delta$ Now you know why $e=3$ was replaced with $e=65,537$.

