Public Key Cryptography: RSA

September 30, 2019

From *The Economist* Sept 15, 2018, page 34

Article Title: Whack a Mole: The new president (of Colombia) calls off talks with a lesser-known leftist insurgent group.

From *The Economist* Sept 15, 2018, page 34

Article Title: Whack a Mole: The new president (of Colombia) calls off talks with a lesser-known leftist insurgent group.

Context: In 2016 FARC, a left-wing insurgent group in Columbia, signed a peace treaty that ended 50 years of conflict Yeah! The former president of Columbia got the Nobel Peace Prize (the leader of FARC did not – I do not know why). However a more extreme insurgent group, ELN, is still active. Why did FARC negotiate but ELN did not?:

From *The Economist* Sept 15, 2018, page 34

Article Title: Whack a Mole: The new president (of Colombia) calls off talks with a lesser-known leftist insurgent group.

Context: In 2016 FARC, a left-wing insurgent group in Columbia, signed a peace treaty that ended 50 years of conflict Yeah! The former president of Columbia got the Nobel Peace Prize (the leader of FARC did not – I do not know why). However a more extreme insurgent group, ELN, is still active. Why did FARC negotiate but ELN did not?:

Quote: . . . And the ELN's strong *encryption system* has prevented the army from extracting information from seized computers, as it did with FARC.

From *The Economist* Sept 15, 2018, page 34

Article Title: Whack a Mole: The new president (of Colombia) calls off talks with a lesser-known leftist insurgent group.

Context: In 2016 FARC, a left-wing insurgent group in Columbia, signed a peace treaty that ended 50 years of conflict Yeah! The former president of Columbia got the Nobel Peace Prize (the leader of FARC did not – I do not know why). However a more extreme insurgent group, ELN, is still active. Why did FARC negotiate but ELN did not?:

Quote: ... And the ELN's strong *encryption system* has prevented the army from extracting information from seized computers, as it did with FARC.

Caveat: The article did not say what system they used. Oh Well

The Academic Code

September 30, 2019

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

When Academics Says: . . . of great theoretical and practical importance.

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

When Academics Says: . . . of great theoretical and practical importance.
They Mean: interesting to me.

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

When Academics Says: . . . of great theoretical and practical importance.
They Mean: interesting to me.

When Academics Says: It has long been known that. . . .

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

When Academics Says: . . . of great theoretical and practical importance.
They Mean: interesting to me.

When Academics Says: It has long been known that. . . .
They Mean: I haven't bothered to look up the original reference.

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

When Academics Says: . . . of great theoretical and practical importance.
They Mean: interesting to me.

When Academics Says: It has long been known that. . . .
They Mean: I haven't bothered to look up the original reference.

When Academics Says: The proof is left to the reader.

The Academic Code

Academics often talk in code that sounds like normal speech, so you might not realize it. They talk in public, so this could be called public key cryptography.

When Academics Says: . . . of great theoretical and practical importance.
They Mean: interesting to me.

When Academics Says: It has long been known that. . . .
They Mean: I haven't bothered to look up the original reference.

When Academics Says: The proof is left to the reader.
They Mean: Someone smarter than me can surely prove this.

The Academic Code, More Examples

When Academics Says: The agreement of my theory and the empirical data is is Excellent.

The Academic Code, More Examples

When Academics Says: The agreement of my theory and the empirical data is is Excellent.
They Mean: The agreement of my theory and the empirical data is is Good.

The Academic Code, More Examples

When Academics Says: The agreement of my theory and the empirical data is is Excellent.
They Mean: The agreement of my theory and the empirical data is is Good.

When Academics Says: The agreement of my theory and the empirical data is is Good.

The Academic Code, More Examples

When Academics Says: The agreement of my theory and the empirical data is is Excellent.
They Mean: The agreement of my theory and the empirical data is is Good.

When Academics Says: The agreement of my theory and the empirical data is is Good.
They Mean: The agreement of my theory and the empirical data is is Non-existant.

The Academic Code, More Examples

When Academics Says: The agreement of my theory and the empirical data is is Excellent.
They Mean: The agreement of my theory and the empirical data is is Good.

When Academics Says: The agreement of my theory and the empirical data is is Good.
They Mean: The agreement of my theory and the empirical data is is Non-existant.

When Academics Says: It is generally believed that. . .

The Academic Code, More Examples

When Academics Says: The agreement of my theory and the empirical data is is Excellent.
They Mean: The agreement of my theory and the empirical data is is Good.

When Academics Says: The agreement of my theory and the empirical data is is Good.
They Mean: The agreement of my theory and the empirical data is is Non-existant.

When Academics Says: It is generally believed that. . ..
They Mean: Me and my friends think. . ..

Public Key Cryptography: RSA

September 30, 2019

Exponentiation Mod p Revisited

Recall If p prime, $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$.

Exponentiation Mod p Revisited

Recall If p prime, $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$.

How to compute $3^{1000} \pmod{7}$?

Could do repeated squaring. Can we do better? Discuss.

Exponentiation Mod p Revisited

Recall If p prime, $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$.

How to compute $3^{1000} \pmod{7}$?

Could do repeated squaring. Can we do better? Discuss. Yes
By Recall with $p = 7$ and $a = 3$ we have

$$3^6 \equiv 1 \pmod{7}.$$

Exponentiation Mod p Revisited

Recall If p prime, $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$.

How to compute $3^{1000} \pmod{7}$?

Could do repeated squaring. Can we do better? Discuss. Yes

By Recall with $p = 7$ and $a = 3$ we have

$$3^6 \equiv 1 \pmod{7}.$$

$$3^{6k} \equiv (3^6)^k \equiv 1^k \equiv 1.$$

Exponentiation Mod p Revisited

Recall If p prime, $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$.

How to compute $3^{1000} \pmod{7}$?

Could do repeated squaring. Can we do better? Discuss. Yes
By Recall with $p = 7$ and $a = 3$ we have

$$3^6 \equiv 1 \pmod{7}.$$

$$3^{6k} \equiv (3^6)^k \equiv 1^k \equiv 1.$$

so

$$3^{1000} \equiv 3^{6 \times 166 + 4} \equiv (3^6)^{166} \times 3^4 \equiv 3^4$$

Exponentiation Mod p Revisited: Another Example

$$11^{999{,}999{,}999} \pmod{107}$$

Repeated squaring would take at least $\lg(999,999,999) \sim 30\ \times$'s.

Exponentiation Mod p Revisited: Another Example

$$11^{999{,}999{,}999} \pmod{107}$$

Repeated squaring would take at least $\lg(999,999,999) \sim 30\ \times$'s.
By Fermats Little Theorem $11^{106} \equiv 1 \pmod{107}$.
Divide 999,999,999 by 106:

$$999,999,999 = 106k + 27 \text{ (don't care what } k \text{ is)}$$

Exponentiation Mod p Revisited: Another Example

$$11^{999,999,999} \pmod{107}$$

Repeated squaring would take at least $\lg(999,999,999) \sim 30$ $\times$'s.
By Fermats Little Theorem $11^{106} \equiv 1 \pmod{107}$.
Divide 999,999,999 by 106:

$$999,999,999 = 106k + 27 \text{ (don't care what } k \text{ is)}$$

$$11^{999,999,999} = 11^{106k} \times 11^{27} = (11^{106})^k \equiv 1^k 11^{27} \equiv 11^{27} \pmod{107}$$

Exponentiation Mod p Revisited: Another Example

$$11^{999{,}999{,}999} \pmod{107}$$

Repeated squaring would take at least $\lg(999,999,999) \sim 30$ $\times$'s.
By Fermats Little Theorem $11^{106} \equiv 1 \pmod{107}$.
Divide 999,999,999 by 106:

$$999,999,999 = 106k + 27 \text{ (don't care what } k \text{ is)}$$

$$11^{999{,}999{,}999} = 11^{106k} \times 11^{27} = (11^{106})^k \equiv 1^k 11^{27} \equiv 11^{27} \pmod{107}$$

Lets rewrite that

$$11^{999{,}999{,}999} \equiv 11^{999{,}999{,}999 \pmod{106}} \pmod{107} \equiv 11^{27} \pmod{107}$$

Now do normal repeated squaring. 10 $\times$'s total.

Exponentiation Mod p Revisited: Another Example

$$11^{999{,}999{,}999} \pmod{107}$$

Repeated squaring would take at least $\lg(999,999,999) \sim 30$ $\times$'s.
By Fermats Little Theorem $11^{106} \equiv 1 \pmod{107}$.
Divide 999,999,999 by 106:

$$999,999,999 = 106k + 27 \text{ (don't care what } k \text{ is)}$$

$$11^{999{,}999{,}999} = 11^{106k} \times 11^{27} = (11^{106})^k \equiv 1^k 11^{27} \equiv 11^{27} \pmod{107}$$

Lets rewrite that

$$11^{999{,}999{,}999} \equiv 11^{999{,}999{,}999 \pmod{106}} \pmod{107} \equiv 11^{27} \pmod{107}$$

Now do normal repeated squaring. 10 $\times$'s total.
Can we generalize?

Exponentiation Mod p Revisited: Another Example

$$11^{999{,}999{,}999} \pmod{107}$$

Repeated squaring would take at least $\lg(999,999,999) \sim 30 \ \times$'s.
By Fermats Little Theorem $11^{106} \equiv 1 \pmod{107}$.
Divide 999,999,999 by 106:

$$999,999,999 = 106k + 27 \text{ (don't care what } k \text{ is)}$$

$$11^{999{,}999{,}999} = 11^{106k} \times 11^{27} = (11^{106})^k \equiv 1^k 11^{27} \equiv 11^{27} \pmod{107}$$

Lets rewrite that

$$11^{999{,}999{,}999} \equiv 11^{999{,}999{,}999 \pmod{106}} \pmod{107} \equiv 11^{27} \pmod{107}$$

Now do normal repeated squaring. 10 $\times$'s total.
Can we generalize? Yes

Exponentiation with Really Big Exponents

Generalize p prime, $a \not\equiv 0 \pmod{p}$, $m \in \mathbb{N}$.
We want to compute a^m.

We know that $a^{p-1} \equiv 1 \pmod{p}$.
Divide m by $p-1$:

$m = k(p-1) + r$ where $0 \le r \le p-2$ and $r \equiv m \pmod{p-1}$.
Hence:

$$a^m \equiv a^{k(p-1)+r} \equiv (a^{p-1})^k \times a^r \equiv 1^k a^r \equiv a^r$$

But recall that $r \equiv m \pmod{p-1}$. So

$$a^m \equiv a^{m \bmod p-1} \pmod{p}$$

This last equation is the important point

Needed Mathematics- The ϕ Function

Next few slides are on the ϕ function.

YES, you have already seen it.

Who first said
Math is best learned twice... at least twice.

Needed Mathematics- The ϕ Function

Next few slides are on the ϕ function.

YES, you have already seen it.

Who first said
Math is best learned twice... at least twice.
My CMSC 452 class thought either Gauss or Gasarch.

Needed Mathematics- The ϕ Function

Next few slides are on the ϕ function.

YES, you have already seen it.

Who first said
Math is best learned twice... at least twice.
My CMSC 452 class thought either Gauss or Gasarch.

Answer: Said by Larry Denenberg, who was a grad student in CS the same time Bill Gasarch was.

Needed Mathematics- The ϕ Function

Next few slides are on the ϕ function.

YES, you have already seen it.

Who first said
Math is best learned twice... at least twice.
My CMSC 452 class thought either Gauss or Gasarch.

Answer: Said by Larry Denenberg, who was a grad student in CS the same time Bill Gasarch was. Popularized by Bill Gasarch.

Needed Mathematics- The ϕ Function

Next few slides are on the ϕ function.

YES, you have already seen it.

Who first said
Math is best learned twice... at least twice.
My CMSC 452 class thought either Gauss or Gasarch.

Answer: Said by Larry Denenberg, who was a grad student in CS the same time Bill Gasarch was. Popularized by Bill Gasarch. Probably not said by Gauss.

Needed Mathematics- The ϕ Function

Next few slides are on the ϕ function.

YES, you have already seen it.

Who first said
Math is best learned twice... at least twice.
My CMSC 452 class thought either Gauss or Gasarch.

Answer: Said by Larry Denenberg, who was a grad student in CS the same time Bill Gasarch was. Popularized by Bill Gasarch. Probably not said by Gauss. Probably not true for Gauss.

Needed Mathematics- The ϕ Function

Recall If p is prime and $1 \le a \le p-1$ then $a^{p-1} \equiv 1 \pmod{p}$.
Recall: For all m, $a^m \equiv a^{m \pmod{p-1}} \pmod{p}$.
So arithmetic in the exponents is mod $p-1$.

We need to generalize this to when the mod is not a prime.

Definition
$\phi(n)$ is the number of numbers in $\{1, \ldots, n\}$ that are relatively prime to n.

Recall: If p is prime then $\phi(p) = p - 1$.
Recall: If a, b rel prime then $\phi(ab) = \phi(a)\phi(b)$.

Theorem for Primes, Theorem for n

We restate and generalize.

Fermat's Little Theorem: If p is prime and $a \not\equiv 0 \pmod{p}$ then

$$a^m \equiv a^{m \bmod p-1} \pmod{p}.$$

Restate:

Fermat's Little Theorem: If p is prime and a is rel prime to p then

$$a^m \equiv a^{m \bmod p-1} \pmod{p}.$$

Generalize:

Fermat-Euler Theorem: If $n \in \mathbb{N}$ and a is rel prime to n then

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}.$$

Examples

$$14^{999,999} \pmod{393}$$

$$\phi(393) = \phi(3 \times 131) = \phi(3) \times \phi(131) = 2 \times 130 = 260.$$

$$14^{999,999} = 14^{999,999 \pmod{260}} \pmod{393} \equiv 14^{39} \pmod{393}$$

Now just do repeated squaring.

Bait and Switch

I got you interested in the theorem

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}$$

by telling you that it can be used to do things like

$$17^{191{,}992{,}194{,}299{,}292777} \pmod{150}.$$

with FAR less than $2\lg(191, 992, 194, 299, 292777)$ $\times$'s.

Bait and Switch

I got you interested in the theorem

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}$$

by telling you that it can be used to do things like

$$17^{191,992,194,299,292777} \pmod{150}.$$

with FAR less than $2\lg(191,992,194,299,292777)$ $\times$'s.
This is true! There will be some HW using it.

You are thinking:

Bait and Switch

I got you interested in the theorem

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}$$

by telling you that it can be used to do things like

$$17^{191{,}992{,}194{,}299{,}292777} \pmod{150}.$$

with FAR less than $2\lg(191, 992, 194, 299, 292777)$ $\times$'s.
This is true! There will be some HW using it.

You are thinking: A&B will need to do $a^m \pmod{n}$ for large m.

Bait and Switch

I got you interested in the theorem

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}$$

by telling you that it can be used to do things like

$$17^{191,992,194,299,292777} \pmod{150}.$$

with FAR less than $2\lg(191, 992, 194, 299, 292777)$ $\times$'s.
This is true! There will be some HW using it.

You are thinking: A&B will need to do $a^m \pmod{n}$ for large m.

No. That is not what we will be doing, though I see why you would think that. Or you see why I think you would think that. Or ....

Bait and Switch

I got you interested in the theorem

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}$$

by telling you that it can be used to do things like

$$17^{191,992,194,299,292777} \pmod{150}.$$

with FAR less than $2\lg(191,992,194,299,292777)$ $\times$'s.
This is true! There will be some HW using it.

You are thinking: A&B will need to do $a^m \pmod{n}$ for large m.

No. That is not what we will be doing, though I see why you would think that. Or you see why I think you would think that. Or ....
We will just use the theorem:

$$a^m \equiv a^{m \bmod \phi(n)} \pmod{n}$$

Easy and Hard

Known to be Easy, Do in Order

1. Given L, generate two primes of length L: p, q.
2. Compute $N = pq$ and $R = (p-1)(q-1)$.
3. Find e rel prime to R.
4. If have p, q then Find d such that $ed \equiv 1 \pmod{R}$. KEY: Easy since have p, q. Would be hard otherwise
5. Compute $m^e \pmod{N}$.

Thought to be Hard

Given N, e as above find d as above. Note that we are not given p, q or R.

RSA

Let L be a security parameter

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)
6. Bob: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)
6. Bob: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.
7. If Alice gets $m^e \pmod{N}$ she computes

$$(m^e)^d \equiv m^{ed} \equiv m^{ed \bmod R} \equiv m^{1 \bmod R} \equiv m$$

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)
6. Bob: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.
7. If Alice gets $m^e \pmod{N}$ she computes

$$(m^e)^d \equiv m^{ed} \equiv m^{ed \bmod R} \equiv m^{1 \bmod R} \equiv m$$

Note: Works $1 \le m \le N-1$. m need not be rel prime to N.

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)
6. Bob: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.
7. If Alice gets $m^e \pmod{N}$ she computes

$$(m^e)^d \equiv m^{ed} \equiv m^{ed \bmod R} \equiv m^{1 \bmod R} \equiv m$$

Note: Works $1 \le m \le N-1$. m need not be rel prime to N.

PRO: Alice and Bob can execute the protocol easily.

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)
6. Bob: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.
7. If Alice gets $m^e \pmod{N}$ she computes

$$(m^e)^d \equiv m^{ed} \equiv m^{ed \bmod R} \equiv m^{1 \bmod R} \equiv m$$

Note: Works $1 \le m \le N-1$. m need not be rel prime to N.

PRO: Alice and Bob can execute the protocol easily.
Biggest PRO: Alice and Bob never had to meet!

RSA

Let L be a security parameter

1. Alice picks two primes p, q of length L and computes $N = pq$.
2. Alice computes $\phi(N) = \phi(pq) = (p-1)(q-1)$. Denote by R
3. Alice picks an $e \in \{\frac{R}{3}, \ldots, \frac{2R}{3}\}$ that is relatively prime to R.
4. Alice finds d such that $ed \equiv 1 \pmod{R}$.
5. Alice broadcasts (N, e). (Bob and Eve both see it.)
6. Bob: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.
7. If Alice gets $m^e \pmod{N}$ she computes

$$(m^e)^d \equiv m^{ed} \equiv m^{ed \bmod R} \equiv m^{1 \bmod R} \equiv m$$

Note: Works $1 \le m \le N-1$. m need not be rel prime to N.

PRO: Alice and Bob can execute the protocol easily.
Biggest PRO: Alice and Bob never had to meet!
Question: Can Eve find out m?

Convention for RSA

Alice sends (N, e) to get the process started

Convention for RSA

Alice sends (N, e) to get the process started

Then Bob can send Alice messages.

Convention for RSA

Alice sends (N, e) to get the process started

Then Bob can send Alice messages.

We don't have Alice sending Bob messages.

Do RSA in Class

Pick out two students to be Alice and Bob.
Use primes
$p = 31$, Prime
$q = 37$, Prime
$N = pq = 31 * 37 = 1147$.
$R = \phi(N) = 30 * 36 = 1080$
$e = 77$, e rel prime to R
$d = 533$ ($ed \equiv 1 \pmod{R}$)
CHECK: $ed = 77 * 533 = 41041 \equiv 1 \pmod{1080}$.
Bob: pick an $m \in \{1, \ldots, N-1\} = \{1, \ldots, 1146\}$. Do not tell us what it is.
Bob: compute $c = m^e \pmod{1147}$ and tell it to us.
Alice: compute $c^d \pmod{1147}$, should get back m.

What Do We Really Know about RSA

If Eve can factor then she can crack RSA.

1. Input (N, e) where $N = pq$ and e is rel prime to $R = (p-1)(q-1)$. (p, q, R are NOT part of the input.)
2. Eve factors N to find p, q. Eve computes $R = (p-1)(q-1)$.
3. Eve finds d such that $ed \equiv 1 \pmod{R}$.

If Factoring Easy then RSA is crackable

What Do We Really Know about RSA

If Eve can factor then she can crack RSA.

1. Input (N, e) where $N = pq$ and e is rel prime to $R = (p-1)(q-1)$. (p, q, R are NOT part of the input.)
2. Eve factors N to find p, q. Eve computes $R = (p-1)(q-1)$.
3. Eve finds d such that $ed \equiv 1 \pmod{R}$.

If Factoring Easy then RSA is crackable

What about converse?

If RSA is crackable then Factoring is Easy

VOTE: TRUE or FALSE or UNKNOWN TO SCIENCE

What Do We Really Know about RSA

If Eve can factor then she can crack RSA.

1. Input (N, e) where $N = pq$ and e is rel prime to $R = (p-1)(q-1)$. (p, q, R are NOT part of the input.)
2. Eve factors N to find p, q. Eve computes $R = (p-1)(q-1)$.
3. Eve finds d such that $ed \equiv 1 \pmod{R}$.

If Factoring Easy then RSA is crackable

What about converse?

If RSA is crackable then Factoring is Easy

VOTE: TRUE or FALSE or UNKNOWN TO SCIENCE
UNKNOWN TO SCIENCE.
Note: In ugrad math classes rare to have a statement that is UNKNOWN TO SCIENCE. Discuss.

Hardness Assumption

Definition

Let f be the following function:
Input: $N, e, m^e \pmod N$ (know $N = pq$ but don't know p, q).

Outputs: m.

Hardness assumption (HA): f is hard to compute.
One can show, assuming HA that RSA is hard to crack. But this proof will depend on a model of security. See caveats about this on similar DH slides (bribery, timing attacks, Maginot Line).

What Could be True?

The following are all possible:

What Could be True?

The following are all possible:

1) Factoring easy. RSA is crackable.

What Could be True?

The following are all possible:

1) Factoring easy. RSA is crackable.

2) Factoring hard, HA false. RSA crackable, Factoring hard!!

What Could be True?

The following are all possible:

1) Factoring easy. RSA is crackable.

2) Factoring hard, HA false. RSA crackable, Factoring hard!!

3) Factoring hard, HA true, but RSA is crackable by other means. Timing Attacks. Must rethink our model of security.

What Could be True?

The following are all possible:

1) Factoring easy. RSA is crackable.

2) Factoring hard, HA false. RSA crackable, Factoring hard!!

3) Factoring hard, HA true, but RSA is crackable by other means. Timing Attacks. Must rethink our model of security.

4) Factoring hard, HA true, and RSA remains uncracked for years. Increases our confidence but ....

What Could be True?

The following are all possible:

1) Factoring easy. RSA is crackable.

2) Factoring hard, HA false. RSA crackable, Factoring hard!!

3) Factoring hard, HA true, but RSA is crackable by other means. Timing Attacks. Must rethink our model of security.

4) Factoring hard, HA true, and RSA remains uncracked for years. Increases our confidence but ....

Item 4 is current state with some caveats: Do Alice and Bob use it properly? Do they have large enough parameters? What is Eve's computing power?

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).
Later Eve sees Bob send Alice c_2 (message is m_2).

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).
Later Eve sees Bob send Alice c_2 (message is m_2).

What can Eve easily deduce?

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).
Later Eve sees Bob send Alice c_2 (message is m_2).

What can Eve easily deduce?

Eve can know if $c_1 = c_2$ or not. So what?

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).
Later Eve sees Bob send Alice c_2 (message is m_2).

What can Eve easily deduce?

Eve can know if $c_1 = c_2$ or not. So what?
Eve knows if $m_1 = m_2$ or not.

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).
Later Eve sees Bob send Alice c_2 (message is m_2).

What can Eve easily deduce?

Eve can know if $c_1 = c_2$ or not. So what?
Eve knows if $m_1 = m_2$ or not.

That alone makes it insecure.

Plain RSA Bytes!

The RSA given above is referred to as Plain RSA.
Insecure!

Scenario:
Eve sees Bob send Alice c_1 (message is m_1).
Later Eve sees Bob send Alice c_2 (message is m_2).

What can Eve easily deduce?

Eve can know if $c_1 = c_2$ or not. So what?
Eve knows if $m_1 = m_2$ or not.

That alone makes it insecure.
Plain RSA is never used and should never be used!

PKCS-1.5 RSA

How can we fix RSA to make it work? Discuss

PKCS-1.5 RSA

How can we fix RSA to make it work? Discuss Need randomness.

PKCS-1.5 RSA

How can we fix RSA to make it work? Discuss Need randomness.

We need to change how Bob sends a message;
BAD: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.

GOOD?: To send $m \in \{1, \ldots, N-1\}$, pick rand r, send $(rm)^e$.
(NOTE- rm means r CONCAT with m here and elsewhere.)

PKCS-1.5 RSA

How can we fix RSA to make it work? Discuss Need randomness.

We need to change how Bob sends a message;
BAD: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.

GOOD?: To send $m \in \{1, \ldots, N-1\}$, pick rand r, send $(rm)^e$.
(NOTE- rm means r CONCAT with m here and elsewhere.)

DEC: Alice can find rm but doesn't know divider. How to fix?

PKCS-1.5 RSA

How can we fix RSA to make it work? Discuss Need randomness.

We need to change how Bob sends a message;
BAD: To send $m \in \{1, \ldots, N-1\}$, send $m^e \pmod{N}$.

GOOD?: To send $m \in \{1, \ldots, N-1\}$, pick rand r, send $(rm)^e$.
(NOTE- rm means r CONCAT with m here and elsewhere.)

DEC: Alice can find rm but doesn't know divider. How to fix?
Alice and Bob agree on dividers ahead of time. Agree on
$L_1 = \left\lfloor \frac{\lg N}{3} \right\rfloor$, $L_2 = \lfloor \lg N \rfloor - L_1$.
To send $m \in \{0,1\}^{L_2}$ pick random $r \in \{0,1\}^{L_1}$.
When Alice gets rm she will know that m is the last L_2 bits.

Example

$p = 31$, Prime $q = 37$, Prime $N = pq = 31 \times 37 = 1147$.
$R = \phi(N) = 30 * 36 = 1080$
$e = 77$ (e rel prime to R), $d = 533$ ($ed \equiv 1 \pmod{R}$)
$L_1 = \left\lfloor \frac{\lg N}{3} \right\rfloor = 3$, $L_2 = \lfloor \lg N \rfloor - L = 7$.
Bob wants to send 1100100 (note- $L_2 = 7$ bits).

1. Bob generates $L_1 = 3$ random bits. 100.
2. Bob sends 1001100100 which is 612 in base 10 by sending $612^{77} \pmod{1147}$ which is 277.
3. Alice decodes by doing $277^{533} \pmod{1147} = 612$
4. Alice puts 612 into binary to get 1001100100. She knows to only read the last 7 bits 1100100.

Important: If later Bob wants to send 100 again he will choose a DIFFERENT random 3 bits so Eve won't know he sent the same message.

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)
Scenario: N and e are public. Bob sends $(rm)^e \pmod{N}$.
Eve cannot determine what m is.

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)
Scenario: N and e are public. Bob sends $(rm)^e \pmod{N}$.
Eve cannot determine what m is.
What can Eve do that is still obnoxious?

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)
Scenario: N and e are public. Bob sends $(rm)^e \pmod{N}$.
Eve cannot determine what m is.
What can Eve do that is still obnoxious?
Eve can compute $2^e(rm)^e \equiv (2(rm))^e \pmod{N}$. So what?

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)
Scenario: N and e are public. Bob sends $(rm)^e \pmod{N}$.
Eve cannot determine what m is.
What can Eve do that is still obnoxious?
Eve can compute $2^e(rm)^e \equiv (2(rm))^e \pmod{N}$. So what?

Eve can later pretend she is Bob and send $(2(rm))^e \pmod{N}$.

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)
Scenario: N and e are public. Bob sends $(rm)^e \pmod{N}$.
Eve cannot determine what m is.
What can Eve do that is still obnoxious?
Eve can compute $2^e(rm)^e \equiv (2(rm))^e \pmod{N}$. So what?

Eve can later pretend she is Bob and send $(2(rm))^e \pmod{N}$.

Why bad? Discuss

Is PKCS-1.5 RSA Secure?

Is PKCS-1.5 RSA Secure? VOTE

- YES (under hardness assumptions and large n)
- NO (there is yet another weird security thing we overlooked)

NO (there is yet another weird security thing we overlooked)
Scenario: N and e are public. Bob sends $(rm)^e \pmod{N}$.
Eve cannot determine what m is.
What can Eve do that is still obnoxious?
Eve can compute $2^e(rm)^e \equiv (2(rm))^e \pmod{N}$. So what?

Eve can later pretend she is Bob and send $(2(rm))^e \pmod{N}$.

Why bad? Discuss
(1) will confuse Alice (2) Sealed Bid Scenario.

Malleability

An encryption system is malleable if when Eve sees a message she can figure out a way to send a similar one, where she knows the similarity (she still does not know the message).

1. The definition above is informal.
2. Can modify RSA so that it's probably not malleable.
3. That way is called PKCS-2.0-RSA.
4. Name BLAH-1.5 is hint that it's not final version.

Final Points About RSA

1. PKCS-2.0-RSA is REALLY used!
2. There are many variants of RSA but all use the ideas above.
3. Factoring easy implies RSA crackable. TRUE.
4. RSA crackable implies Factoring easy: UNKNOWN.
5. RSA crackable implies Factoring easy: Often stated in expositions of crypto. They are wrong!
6. Timing attacks on RSA bypass the math.