Finishing Up Problems with Plain RSA

October 2, 2019
Recall PKCS-1.5 RSA Secure

Plain RSA had NY,NY problem.

We fixed that last lecture.
Is PKCS-1.5 RSA Secure?

VOTE

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

Scenario:

$N$ and $e$ are public. Bob sends $(r^m)^e \pmod{N}$. Eve cannot determine what $m$ is.

What can Eve do that is still obnoxious?

Eve can compute $2^e(r^m)^e \equiv (2(r^m))^e \pmod{N}$. So what?

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

Why bad? Discuss

(1) Alice will think message is $2r^m$.

(2) If the context is money, Alice will think it costs twice as much!
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 $c = m^e \pmod{N}$. Eve cannot determine what $m$ is. What can Eve do that is still obnoxious?

Eve can compute $2^e \cdot c = (2m)^e \equiv (2^e \cdot m)^e \pmod{N}$. So what?

Eve can later pretend she is Bob and send $(2m)^e \pmod{N}$. Why bad? Discuss

1. Alice will think message is $2m$.
2. If the context is money, Alice will think it costs twice as much!
Is PKCS-1.5 RSA Secure?

- 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?

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?

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?

- 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) Alice will think message is $2rm$. (2) If the context is money, Alice will think it costs twice as much!
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 Real 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.
Low $e$ Attacks on RSA
Scenario

1. Zelda is sending messages to Alice using $N_a = 377$, $e = 3$.
2. Zelda is sending messages to Bob using $N_b = 391$, $e = 3$.
3. Zelda is sending messages to Carol using $N_c = 589$, $e = 3$.

$e$ is low. That will make the system crackable if ...
Scenario

1. Zelda is sending messages to Alice using $N_a = 377$, $e = 3$.
2. Zelda is sending messages to Bob using $N_b = 391$, $e = 3$.
3. Zelda is sending messages to Carol using $N_c = 589$, $e = 3$.

$e$ is low. That will make the system crackable if . . .

Zelda sends *same* $m$ to all three. **Note** $m < 377$. Zelda does this:
Scenario

1. Zelda is sending messages to Alice using \( N_a = 377, e = 3 \).
2. Zelda is sending messages to Bob using \( N_b = 391, e = 3 \).
3. Zelda is sending messages to Carol using \( N_c = 589, e = 3 \).

\( e \) is low. That will make the system crackable if . . .

Zelda sends same \( m \) to all three. **Note** \( m < 377 \). Zelda does this:

1. Zelda sends Alice 330. So \( m^3 \equiv 330 \pmod{377} \).
2. Zelda sends Bob 34. So \( m^3 \equiv 34 \pmod{391} \).
3. Zelda sends Carol 419. So \( m^3 \equiv 419 \pmod{589} \).

Eve sees all of this so knows something about \( m \).
1. Zelda is sending messages to Alice using $N_a = 377$, $e = 3$.
2. Zelda is sending messages to Bob using $N_b = 391$, $e = 3$.
3. Zelda is sending messages to Carol using $N_c = 589$, $e = 3$.

$e$ is low. That will make the system crackable if ... Zelda sends *same* $m$ to all three. **Note** $m < 377$. Zelda does this:

1. Zelda sends Alice 330. So $m^3 \equiv 330 \pmod{377}$.
2. Zelda sends Bob 34. So $m^3 \equiv 34 \pmod{391}$.
3. Zelda sends Carol 419. So $m^3 \equiv 419 \pmod{589}$.

Eve sees all of this so knows something about $m$.

We will develop the math and the attack. Called a low-\(e\) attack.
Needed Math: Chinese Remainder Theorem

Example

Find \( x \) such that:

\[
\begin{align*}
  x &\equiv 17 \pmod{31} \\
  x &\equiv 20 \pmod{37}
\end{align*}
\]

a) The inverse of 31 mod 37 is 6
b) The inverse of 37 mod 31 is 26.
c) 

\[
x = 20 \times 6 \times 31 + 17 \times 26 \times 37 = 20,074
\]

\( x \) (mod 31): First term is 0. Second term is 17. So 17.
\( x \) (mod 37): First term is 20. Second term is 0. So 20.
So \( x = 20,074 \) is answer.
Needed Math: Chinese Remainder Theorem

Example

Find $x$ such that:

$$x \equiv 17 \pmod{31} \quad \& \quad x \equiv 20 \pmod{37}$$

So $x = 20,074$ is answer. Can we find a smaller $x$?
We only care about $x \pmod{31}$ and $x \pmod{37}$.

Note:

$$x \equiv 17 \pmod{31} \implies x - 31 \times 37 \equiv 17 \pmod{31}$$
$$x \equiv 20 \pmod{37} \implies x - 31 \times 37 \equiv 20 \pmod{37}$$

If $x$ works then $x - 31 \times 37$ works. So just need

$$20,074 \pmod{31 \times 37} = 575.$$  

Upshot: Can take $x = 20,074 \pmod{31 \times 37} = 575$
What if \( x = m^2 \) is a Square?

Find \( m \) such that:

\[
m^2 \equiv 8 \pmod{17} \quad \& \quad m^2 \equiv 25 \pmod{37}
\]

a) The inverse of 17 mod 37 is 24
b) The inverse of 37 mod 17 is 6

\[
m^2 = 8 \times 37 \times 6 + 25 \times 17 \times 24 = 11976
\]

\[
11976 \equiv 25 \pmod{17 \times 37}.
\]
What if \( x = m^2 \) is a Square?

Find \( m \) such that:

\[
m^2 \equiv 8 \pmod{17} \quad \& \quad m^2 \equiv 25 \pmod{37}
\]

a) The inverse of 17 mod 37 is 24
b) The inverse of 37 mod 17 is 6

\[
m^2 = 8 \times 37 \times 6 + 25 \times 17 \times 24 = 11976
\]

11976 \( \equiv \) 25 \((\text{mod } 17 \times 37)\).

OH, \( m^2 \equiv 25 \). This is a square in \( \mathbb{N} \). So \( m = 5 \).
What if $x = m^3$?

Find $m$ such that:

$$m^3 \equiv 12 \pmod{17} \quad \& \quad m^3 \equiv 16 \pmod{37}$$

a) The inverse of 17 mod 37 is 24
b) The inverse of 37 mod 17 is 6

$$m^3 = 12 \times 37 \times 6 + 16 \times 17 \times 24 = 9192$$

$$9192 \equiv 386 \pmod{17 \times 37}.$$
What if $x = m^3$?

Find $m$ such that:

$$m^3 \equiv 12 \pmod{17} \quad \& \quad m^3 \equiv 16 \pmod{37}$$

a) The inverse of 17 mod 37 is 24
b) The inverse of 37 mod 17 is 6

$$m^3 = 12 \times 37 \times 6 + 16 \times 17 \times 24 = 9192$$

$9192 \equiv 386 \pmod{17 \times 37}$.  
OH, $m^3 \equiv 386$. This is NOT a cube:-(

What was different?
Squares and Cubes

Find $m$ such that:

$$m^2 \equiv 8 \pmod{17} \quad \& \quad m^2 \equiv 25 \pmod{37}$$

The message $m$ is $< 17$ and $< 37$. So

$m^2 < 17 \times 17$. So $m^2 \equiv m^2 \pmod{17 \times 17}$ (no reduce).

We return to this point in a few slides.
Squares and Cubes

Find $m$ such that:

\[ m^2 \equiv 8 \pmod{17} \quad \& \quad m^2 \equiv 25 \pmod{37} \]

The message $m$ is $< 17$ and $< 37$. So $m^2 < 17 \times 17$. So $m^2 \equiv m^2 \pmod{17 \times 17}$ (no reduce).

Find $m$ such that:

\[ m^3 \equiv 12 \pmod{17} \quad \& \quad m^3 \equiv 16 \pmod{37} \]

The message $m$ is $< 17$ and $< 37$, so $m^3 < 17^3 = 4913$. So $m^3 \pmod{17 \times 37}$ CAN reduce. So DO NOT get that

\[ m^3 \pmod{17 \times 37} = m^3 \]
Squares and Cubes

Find $m$ such that:

\[ m^2 \equiv 8 \pmod{17} \quad \& \quad m^2 \equiv 25 \pmod{37} \]

The message $m$ is $< 17$ and $< 37$. So $m^2 < 17 \times 17$. So $m^2 \equiv m^2 \pmod{17 \times 17}$ (no reduce).

Find $m$ such that:

\[ m^3 \equiv 12 \pmod{17} \quad \& \quad m^3 \equiv 16 \pmod{37} \]

The message $m$ is $< 17$ and $< 37$, so $m^3 < 17^3 = 4913$. So $m^3 \pmod{17 \times 37}$ CAN reduce. So DO NOT get that

\[ m^3 \pmod{17 \times 37} = m^3 \]

We return to this point in a few slides.
Needed Math: Chinese Remainder Theorem $N_1, N_2$

Case

1. Input $a, b, N_1, N_2$, with $N_1, N_2$, rel primes. Want $0 \leq x < N_1N_2$:
   
   \[
   x \equiv a \pmod{N_1} \\
   x \equiv b \pmod{N_2}
   \]

2. Find the inverse of $N_1 \bmod N_2$ and denote this $N_1^{-1}$.
3. Find the inverse of $N_2 \bmod N_1$ and denote this $N_2^{-1}$.
4. \( y = bN_1^{-1}N_1 + aN_2^{-1}N_2 \)
   
   Mod $N_1$: 1st term is 0, 2nd term is $a$. So $y \equiv a \pmod{N_1}$.
   
   Mod $N_2$: 2nd term is 0, 1st term is $b$. So $y \equiv b \pmod{N_2}$.

5. $x \equiv y \pmod{N_1N_2}$. (Convention that $0 \leq x < N_1N_2$)
Needed Math: The Chinese Remainder Theorem

**Theorem:** If $N_1, \ldots, N_L$ are rel prime, $x_1, \ldots, x_L$ are anything, then there exists $x$ with $0 \leq x < N_1 \cdots N_L$ such that

\[ x \equiv x_1 \pmod{N_1} \]

\[ x \equiv x_2 \pmod{N_2} \]

\[ \vdots \]

\[ x \equiv x_L \pmod{N_L} \]

**Proof:** Omitted.

**Notation:** CRT is Chinese Remainder Theorem.
Needed Math: The e Theorem, $N_1, N_2$ case

**Theorem:** Assume $N_1, N_2$ are rel prime, $e, m \in \mathbb{N}$. Let $0 \leq x < N_1N_2$ be the number from CRT such that
\[ x \equiv m^e \pmod{N_1} \]
\[ x \equiv m^e \pmod{N_2} \]
Then $x \equiv m^e \pmod{N_1N_2}$. **IF** $m^e < N_1N_2$ **then** $x = m^e$.

**Proof:** There exists $k_1, k_2$ such that
\[ x = m^e + k_1N_1 \quad k_1 \in \mathbb{Z}, \text{ (Could be negative)} \]
\[ x = m^e + k_2N_2 \quad k_2 \in \mathbb{Z}, \text{ (Could be negative)} \]

$k_1N_1 = k_2N_2$. Since $N_1, N_2$ rel prime, $N_1$ divides $k_2$, so $k_2 = kN_1$. $x = m^e + kN_1N_2$. Hence $x \equiv m^e \pmod{N_1N_2}$.

If $m^e < N_1N_2$ then since $0 \leq x < N_1N_2 \& x \equiv m^e$, $x = m^e$. 
Needed Math: The $e$ Theorem, $N_1, \ldots, N_L$ Case

**Theorem:** Assume $N_1, \ldots, N_L$ are rel prime, $e, m \in \mathbb{N}$.

\[ x \equiv m^e \pmod{N_1} \]
\[ \vdots \]
\[ x \equiv m^e \pmod{N_L} \]

Then $x \equiv m^e \pmod{N_1 \cdots N_L}$. If $m^e < N_1 \cdots N_L$ then $x = m^e$.

**Proof:** Omitted.
Using CRT to find \( m \)

**Theorem:** Assume \( N_1, \ldots, N_L \) are rel prime, \( e, m \in \mathbb{N} \), \( e \leq L \), and for all \( i \), \( m < N_i \). Assume you are given, for all \( i \), \( x_i \) such that \( m^e \equiv x_i \pmod{N_i} \) (you are NOT given \( m \)). Then you can find \( m \).

**Proof:** Use CRT to find \( x \) such that

\[
\begin{align*}
x &\equiv x_1 \pmod{N_1} \\
& \vdots \qquad \vdots \\
x &\equiv x_L \pmod{N_L}
\end{align*}
\]

and \( 0 \leq x < N_1 \cdots N_L \).

Since \( m < N_i \) and \( e \leq L \), \( m^e < N_1 \cdots N_L \).

Hence \( x \) is an \( e \)th power in \( \mathbb{N} \). Take the \( e \)th root to find \( m \).

End of Proof
Low Exponent Attack: Example

1) \(N_a = 377, N_b = 391, N_c = 589\). For Alice, Bob, Carol.
2) \(e = 3\).
3) Zelda sends \(m\) to all three. Eve will find \(m\). Note \(m < 377\).
   1. Zelda sends Alice 330. So \(m^3 \equiv 330 \pmod{377}\).
   2. Zelda sends Bob 34. So \(m^3 \equiv 34 \pmod{391}\).
   3. Zelda sends Carol 419. So \(m^3 \equiv 419 \pmod{589}\).

Eve sees all of this. Eve uses CRT to find \(0 \leq x < 377 \times 391 \times 589\).
\[
x \equiv 330 \equiv m^3 \pmod{377}
\]
\[
x \equiv 34 \equiv m^3 \pmod{391}
\]
\[
x \equiv 419 \equiv m^3 \pmod{589}
\]
Eve finds such a number: \(x = 1,061,208\).

By \(e\)-Theorem

\[
1,061,208 \equiv m^3 \pmod{377 \times 391 \times 589}.
\]
Low Exponent Attack: Example Continued

By e-Theorem

$$1,061,208 \equiv m^3 \pmod{377 \times 391 \times 589}.$$ 

**Most Important Fact:** Recall that $m < 377$. Hence note that:

$$m^3 < 377 \times 377 \times 377 < 377 \times 391 \times 589$$

$$m^3 \equiv 1,061,208 \pmod{377 \times 391 \times 589}$$

Therefore the $m^3$ calculation cannot have wrap-around. Hence $m$ can be gotten from the ordinary cube root operation. We find

$$(1,061,208)^{1/3} = 102$$

So $m = 102$,

**Note:** Cracked RSA without factoring.
Where Did $e = 3$ Come Into This?

Since $m < 377$ we had:

$$m^3 < 377 \times 377 \times 377 < 377 \times 391 \times 589$$

What if $e = 4$? Then everything goes through until we get to:

$$m^4 < 377 \times 377 \times 377 \times 377$$

We need this to be $< 377 \times 391 \times 589$.

But it’s not. So we needed

$$e \leq \text{The number of people}$$
Low Exponent Attack: Generalized

1) $L$ people. Use $N_1 < \cdots < N_L$. All Rel Prime.
2) $e \leq L$
3) Zelda sends $m$ to $L$ people. Note $m < N_1$. 

---

4) You will finish this on HW. You will write pseudocode.
Can you run the algorithm even if $e$ is not small? Discuss
Yes Run it and if $me < N_1 \cdots N_L$ then will still work. You will know it doesn't work if when you need to find an $e$th root (in $N$) there is none (in $N$).
Low Exponent Attack: Generalized

1) $L$ people. Use $N_1 < \cdots < N_L$. All Rel Prime.
2) $e \leq L$
3) Zelda sends $m$ to $L$ people. Note $m < N_1$.
4) You will finish this on HW. You will write pseudocode.

Can you run the algorithm even if $e$ is not small? Discuss
Low Exponent Attack: Generalized

1) $L$ people. Use $N_1 < \cdots < N_L$. All Rel Prime.
2) $e \leq L$
3) Zelda sends $m$ to $L$ people. Note $m < N_1$.
4) You will finish this on HW. You will write pseudocode.

Can you run the algorithm even if $e$ is not small? Discuss

Yes Run it and if $m^e < N_1 \cdots N_L$ then will still work. You will know it doesn't work if when you need to find an $e$th root (in $\mathbb{N}$) there is none (in $\mathbb{N}$).