

Modular arithmetic and trap door ciphers



Prof. Joshua Holden, Rose-Hulman Inst. of Tech.

<http://www.rose-hulman.edu/~holden>

RSA Setup

Ronald Rivest, Adi Shamir, Leonard Adleman,
1977.

- Pick two primes p and q .
- Compute $n = pq$.
- Pick *encryption exponent* e such that e and $(p - 1)(q - 1)$ don't have any common prime factors.
- Make n and e public. Keep p and q private.

RSA Setup: Example

- $p = 53$
- $q = 71$
- $n = pq = 3763$
- $(p - 1)(q - 1) = 2^3 \cdot 5 \cdot 7 \cdot 13$
- $e = 27 = 3^3$
- e and $(p - 1)(q - 1)$ don't have any common prime factors

RSA Setup: PGP public key block

From holden@math.duke.edu Thu Feb 8 14:07:19 2001

Date: Thu, 8 Feb 2001 14:07:18 -0500

X-Authentication-Warning: hamburg.math.duke.edu: holden set sender to holden

From: Joshua Holden To: holden@math.duke.edu

Subject: message with PGP block

Here is my PGP block: now you can send me messages!

-----BEGIN PGP PUBLIC KEY BLOCK-----

Version: 2.6.2

Comment: Processed by Mailcrypt 3.5.5, an Emacs/PGP interface

```
mQCNAznRHaMAAAEEAPix/FD/jF/ixMvd9aIjhZ/K6o2kv/TaGAVkeIG5VZ48jzIa
NTqX1EKDw6aABUiQApqavOaQuiLbi/Ez9HXX9LfcTdcP8u94BKGgmEy6Jv1za08I
2YVL1kUJso6lauryr3Sc8wiQTwx3imohM4ai/1dVuq4Qp2WCBSRdyaafdchdAAUR
tC9Kb3NodWEgSG9sZGVuICgxMDI0IGJpdCkgPGhvbGR1bkBtYXRoLmR1a2UuZWR1
Pg==
```

=VgE9

-----END PGP PUBLIC KEY BLOCK-----

Modular Arithmetic

Karl Friedrich Gauss, 1801.

- Modular Arithmetic = "Wrap-around" computations
- *Example:* Start at 12 o'clock. 5 hours plus 8 hours equals 1 o'clock.
- $5 + 8 \equiv 1 \pmod{12}$
- *Example:* Start at 12 o'clock. 11 hours times 5 equals 7 o'clock.
- $11 \cdot 5 \equiv 7 \pmod{12}$

RSA Encryption

- Anyone can encrypt, because n and e are public.
- To encrypt, convert your message into a set of *plaintext* numbers P , each less than n .
- For each P , compute $C \equiv P^e \pmod{n}$.
- The numbers C are your *ciphertext*.

RSA Encryption: Example

Send the message "cats and dogs":

- ca ts an dd og sx
- 0200 1918 0013 0303 1406 1823
- $200^e \equiv 12 \pmod{n}$
- $1918^e \equiv 1918 \pmod{n}$
- $13^e \equiv 1550 \pmod{n}$
- $303^e \equiv 3483 \pmod{n}$
- $1406^e \equiv 2042 \pmod{n}$
- $1823^e \equiv 2735 \pmod{n}$

RSA Encryption: PGP message

From holden@math.duke.edu Thu Feb 8 14:09:25 2001

Date: Thu, 8 Feb 2001 14:09:24 -0500

X-Authentication-Warning: hamburg.math.duke.edu: holden set sender to holden

From: Joshua Holden To: holden@math.duke.edu

Subject: This message is encrypted

-----BEGIN PGP MESSAGE-----

Version: 2.6.2

Comment: Processed by Mailcrypt 3.5.5, an Emacs/PGP interface

hIwDJF3Jpp91yF0BBAC6gnKTMhGWg9hUeLd7WfJgUn7OqObCNmvm9V8ff+tyq0re
nSQqCYw784CAkm5gaUtJ0AW4go2pDyI2hm5ocoVfMeBOJpKeckSchncV9zHSH2zx
jBM8W0NYPAAa7AHFisz19rqxkkt1aQ4W49i7LUxq6rXheoSPMMcHbHyBa/mQEaYA
AABEmtEXwkUSMOh+x4dSM/6ZUswVZznmei9TOw+md8OM+LiOSakw91GT431tJPAN
c44q+q2Yq8ehykaz0sV4fXscPy2H9A0=
=v1z0

-----END PGP MESSAGE-----

Trap Door

Leonhard Euler, 1736.

- Let $\phi(n)$ be the number of positive integers less than or equal to n which don't have any common factors with n .
- *Example:* If $n = 15$, then the positive integers less than or equal to n which don't have any common factors with n are 1, 2, 4, 7, 8, 11, 13, 14. So $\phi(15) = 8$.

Trap Door: RSA

- In the RSA system $n = pq$, so $\phi(n)$ is the number of positive integers less than or equal to n which don't have p or q as a factor.
- How many positive integers less than or equal to n do have p as a factor? $p, 2p, 3p, \dots, n = qp$ so there are q of them.
- Similarly, there are p positive integers less than or equal to n with q as a factor.
- Only one positive integer less than or equal to n has both p and q as factors, namely $n = pq$. So we should only count this once.

Trap Door: Formula

- Therefore, $\phi(n) = n - p - q + 1 = pq - p - q - 1 = (p - 1)(q - 1)$.
- This is private! You can't calculate it without knowing p and q .
- Why is this useful?

Euler's Theorem

- *Euler's Theorem:* If x is an integer which has no common prime factors with n , then

$$x^{\phi(n)} \equiv 1 \pmod{n}.$$

- Why is Euler's Theorem true?
- Number Theory idea: We consider the positive integers less than or equal to n which don't have any common factors with n , and multiply each of them by x modulo n . Compare them to the same integers without multiplying by x .

Euler's Theorem: Example (I)

- For $n = 15$, consider

$$x, 2x, 4x, 7x, 8x, 11x, 13x, 14x \pmod{15},$$

and compare them to 1, 2, 4, 7, 8, 11, 13, 14.

- If we multiply all of the first set we get

$$x^8 \cdot 1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \pmod{15}$$

and if we multiply all of the second set we get

$$1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \pmod{15}.$$

- What if we do all of this for $x = 11$?

Euler's Theorem: Example (II)

- The first set will be
- $1 \cdot 11 \equiv 11 \pmod{15}$
- $2 \cdot 11 \equiv 7 \pmod{15}$
- $4 \cdot 11 \equiv 14 \pmod{15}$
- $7 \cdot 11 \equiv 2 \pmod{15}$
- $8 \cdot 11 \equiv 13 \pmod{15}$
- $11 \cdot 11 \equiv 1 \pmod{15}$
- $13 \cdot 11 \equiv 8 \pmod{15}$
- $14 \cdot 11 \equiv 4 \pmod{15}$

Euler's Theorem: Example (III)

- The first set is the same as the second set, only in a different order!
- In fact, this always happens.
- So

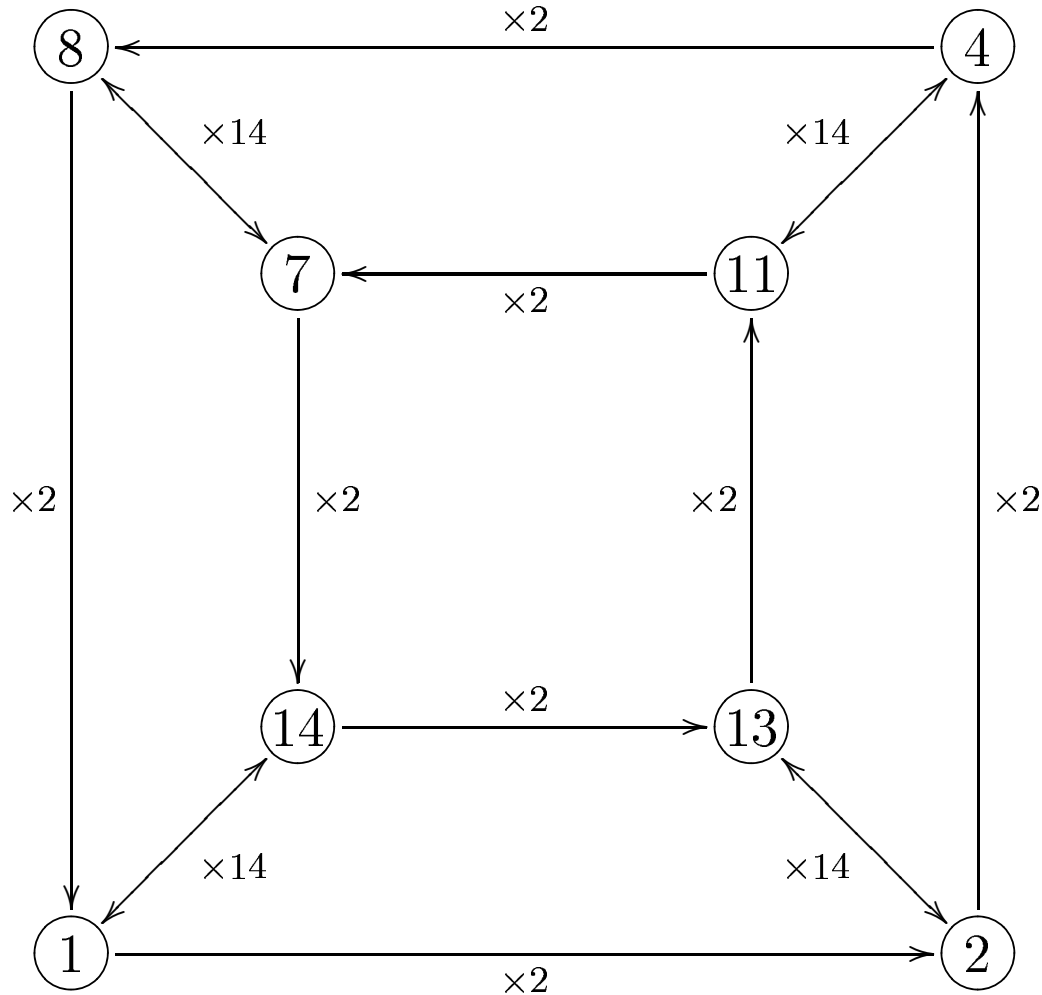
$$x^8 \cdot 1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \equiv 1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \pmod{15}$$

or

$$x^8 \equiv 1 \pmod{15}.$$

Cayley diagram

Arthur Cayley, 1878.



- Group Theory idea: We make a *Cayley diagram* for the numbers less than n .

Cayley diagram: Example (II)

- Say $x = 11$. Follow the arrows from 1 to 11. This is one $\times 14$ arrow and two $\times 2$ arrows. If you do this 7 more times, you will be following a total of eight $\times 14$ arrows and sixteen $\times 2$ arrows, and you should end up at 11 to the eighth. However, eight $\times 14$ arrows and sixteen $\times 2$ arrows clearly ends you up back where you started! (Note that it doesn't matter in what order you follow the arrows....)
- So how do we use the trap door?

RSA: One More Piece

- We need one more piece of (private) information.

Euclid, about 300 B.C.E.

- If a and b don't have any common prime factors, then there are integers c and d such that

$$ac + bd = 1.$$

Euclidean Algorithm

- Since we picked e such that e and $(p - 1)(q - 1)$ don't have any common prime factors, then there are integers c and d such that

$$(p - 1)(q - 1)c + ed = 1$$

or

$$\phi(n)c + ed = 1.$$

- Euclid also tells us how to find c and d , using the *Euclidean Algorithm*.
- Once we have found the *decryption exponent* d , which is private, we can decrypt.

RSA Decryption

- For each C , compute $C^d \pmod{n}$.
- What will this give you?
- We know $C \equiv P^e \pmod{n}$, although we don't yet know what P is. So

$$C^d \equiv (P^e)^d \equiv P^{ed} \equiv P^{1-\phi(n)c} \equiv P(P^{\phi(n)})^{-c} \pmod{n}.$$

- But $P^{\phi(n)} \equiv 1 \pmod{n}$ by Euler's Theorem!
- So $C^d \equiv P \pmod{n}$ and we get our original plaintext back.

RSA Decryption: Example (I)

- $p = 53$
- $q = 71$
- $(p - 1)(q - 1) = 3640$
- $e = 27$
- The Euclidean Algorithm tells us

$$16(p - 1)(q - 1) - 2157e = 1.$$

- $d = -2157$

RSA Decryption: Example (II)

- $12^d \equiv 200 \pmod{n}$
- $1918^d \equiv 1918 \pmod{n}$
- $1550^d \equiv 13 \pmod{n}$
- $3483^d \equiv 303 \pmod{n}$
- $2042^d \equiv 1406 \pmod{n}$
- $2735^d \equiv 1823 \pmod{n}$
- 0200 1918 0013 0303 1406 1823
- ca ts an dd og sx
- “cats and dogs”

Breaking RSA: Factoring

- So why do we think RSA is secure?
- As far as we know, the only way to break RSA is to find p and q by factoring n . The fastest known factoring algorithm takes something about like

$$e^{(\log n)^{1/3}(\log(\log n))^{2/3}}$$

time units to factor n . (The size of the time unit depends on things like how fast the computer is!)

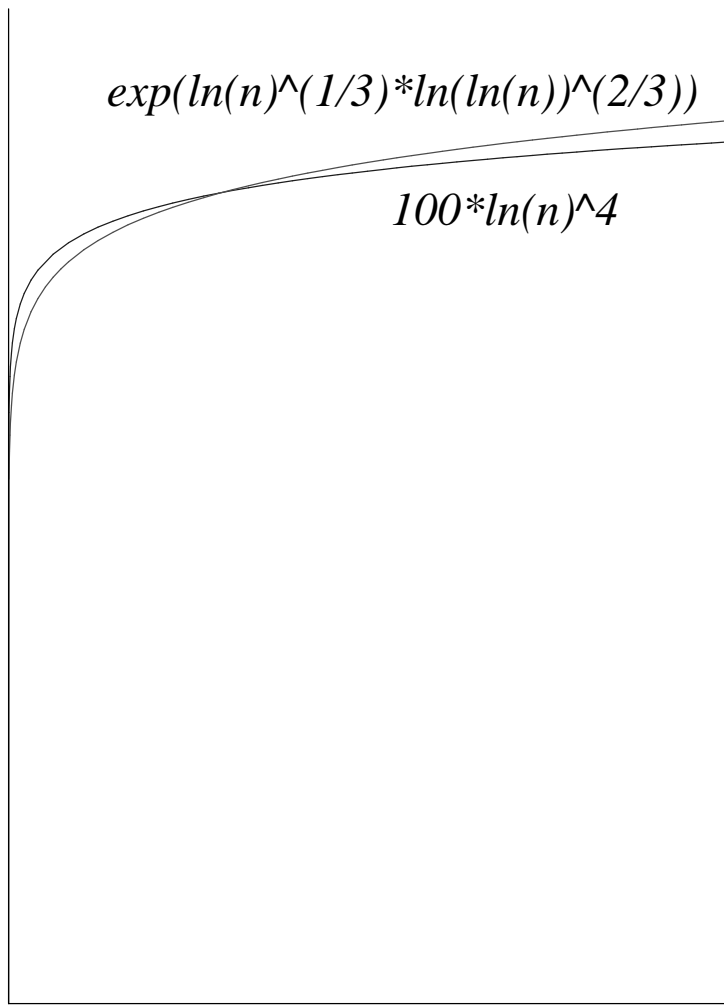
Breaking RSA: Fast computers

- For the fastest single computer in 2006, it would probably take about 1 billion years to factor a number with 300 decimal digits. However, with networked computers, a large company might be able to improve this by a factor of as much as 1 million.
- (More technically, it is estimated that factoring a number with 300 decimal digits would take about 10^{11} MIPS-years. 1 MIPS-year is a million-instructions-per-second processor running for one year. A 1-GHz Pentium is about a 250-MIPS machine.)

Breaking RSA: Factoring vs. Setup

- On the other hand, finding p and q and multiplying them together is very fast. Finding a number p which is (probably) prime takes about $100(\log p)^4$ time units. This looks large, but it isn't really; for a 300-digit number this should only take a few minutes. (Multiplying p and q together is even faster.)

Breaking RSA: A Graph



- At some size of n it will always be easier to make the cipher than to break it!

RSA Digital Signatures

- As a bonus, RSA gives us a way to digitally “sign” messages, thereby proving who wrote them. This uses the same public n and e and private d as before.
- For each plaintext P , compute $S \equiv P^d \pmod{n}$.
- The numbers S are your signed message.

RSA Digital Signatures: Example

Sign the message "cats and dogs":

- ca ts an dd og sx
- 0200 1918 0013 0303 1406 1823
- $200^d \equiv 648 \pmod{n}$
- $1918^d \equiv 1918 \pmod{n}$
- $13^d \equiv 914 \pmod{n}$
- $303^d \equiv 1946 \pmod{n}$
- $1406^d \equiv 664 \pmod{n}$
- $1823^d \equiv 2735 \pmod{n}$

RSA Digital Signatures: PGP message

From holden@math.duke.edu Thu Feb 8 14:10:42 2001

Date: Thu, 8 Feb 2001 14:10:41 -0500

X-Authentication-Warning: hamburg.math.duke.edu: holden set sender to holden

From: Joshua Holden To: holden@math.duke.edu

Subject: This message is signed but not encrypted

-----BEGIN PGP SIGNED MESSAGE-----

I'm signing this message so that you know it's me!

-----BEGIN PGP SIGNATURE-----

Version: 2.6.2

Comment: Processed by Mailcrypt 3.5.5, an Emacs/PGP interface

iQCVAwUB0oLvKyRdyaafdchdAQELuQP+PBR2lY8rEPrgA4GzWQS/MbE4UDECKgBk
v+6Q/gAwrHzMwemXcZxKU1FGFClvfHxxyjoy8hJgYeLYiGvD+q1lgtNGZtTdLzqh
xL/Bdw75fseFxa1/32ZS45jMA3gA2220m70hkJg4EzyvlhDUdUI1SIQHn/V26H0g
I25V0m/Ib8s=
=CRW2

-----END PGP SIGNATURE-----

Verifying the Signature

- Since only you know the decryption exponent d , only you can sign a message. Anyone you send it to can prove it was you by computing $S^e \pmod{n}$ (since n and e are public) and getting back $P^{de} \pmod{n}$, which we know is congruent to P .
- If this matches the P which you sent separately, then the message was correctly signed, so it must have come from someone who knows d .

Verifying the Signature: Example

- $648^e \equiv 200 \pmod{n}$
- $1918^e \equiv 1918 \pmod{n}$
- $914^e \equiv 13 \pmod{n}$
- $1946^e \equiv 303 \pmod{n}$
- $664^e \equiv 1406 \pmod{n}$
- $2735^e \equiv 1823 \pmod{n}$
- 0200 1918 0013 0303 1406 1823
- ca ts an dd og sx
- “cats and dogs”