RSA Cryptosystem

MA 623 Number Theory

Kishen N Gowda (17110074)

June 24, 2020

Indian Institute of Technology, Gandhinagar

Introduction

An Encrypted World!

- Cryptography is the study of secure communications techniques that allow only the sender and intended recipient of a message to view its contents.

An Encrypted World!

- Cryptography is the study of secure communications techniques that allow only the sender and intended recipient of a message to view its contents.
- Some examples we see daily
  - Authentication/Digital Signatures
  - Electronic Money (Online Banking, Debit/Credit Cards)
  - Emails, WhatsApp, Sim Card Authentication
  - Laptop, Mobile Encryption
  - Much more..

A Brief History of Cryptography

- Hieroglyphs (Egyptians, 4000 years ago), Caesar Shift Cipher (Roman Empire)

A Hieroglyph [1]

[1] https://www.greatschools.org/gk/articles/making-hieroglyphics/

A Brief History of Cryptography

- Hieroglyphs (Egyptians, 4000 years ago), Caesar Shift Cipher (Roman Empire)

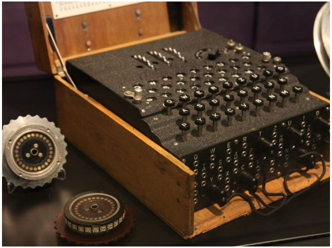
A Hieroglyph [1]

- Vigenere cipher, “The Beale Ciphers”

[1] https://www.greatschools.org/gk/articles/making-hieroglyphics/

A Brief History of Cryptography

- Enigma: World War 2

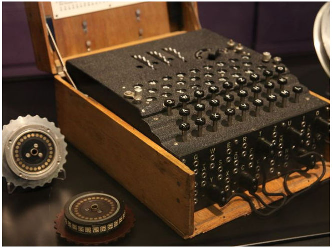


The Enigma machine [1]

[1] https://yorkissp.org/2017/03/13/7-8-lecture-codes-ciphers/

A Brief History of Cryptography

- Enigma: World War 2



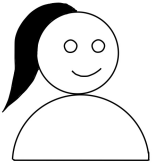
The Enigma machine [1]

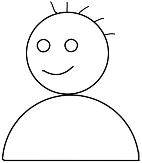
- During World War 2, cryptograpy and cryptanalysis became excessively mathematical.

[1] https://yorkissp.org/2017/03/13/7-8-lecture-codes-ciphers/

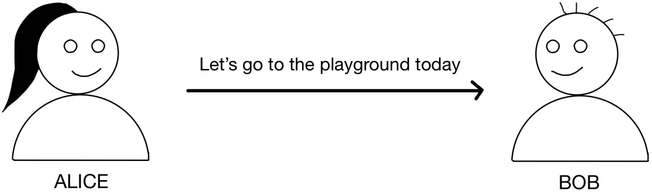
Preliminaries

People with the most secrets!

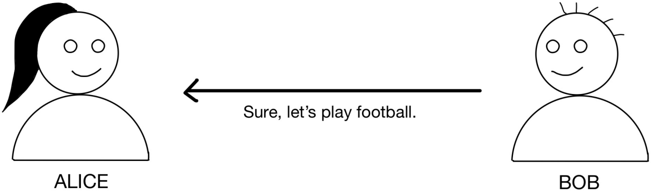




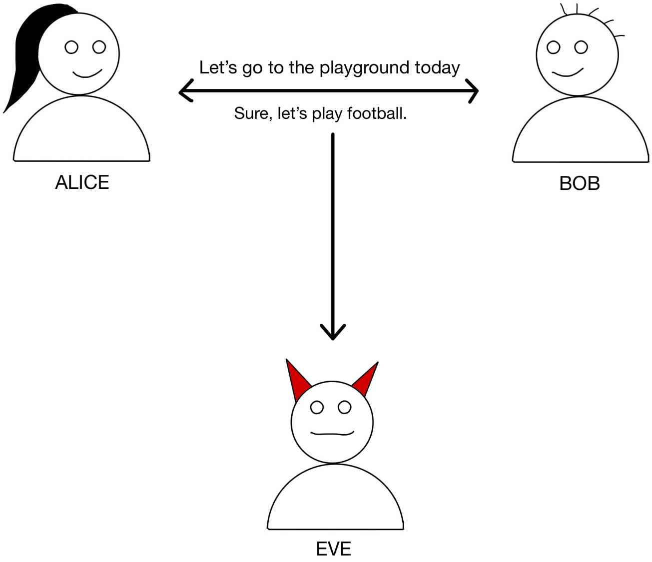
Secure Communication



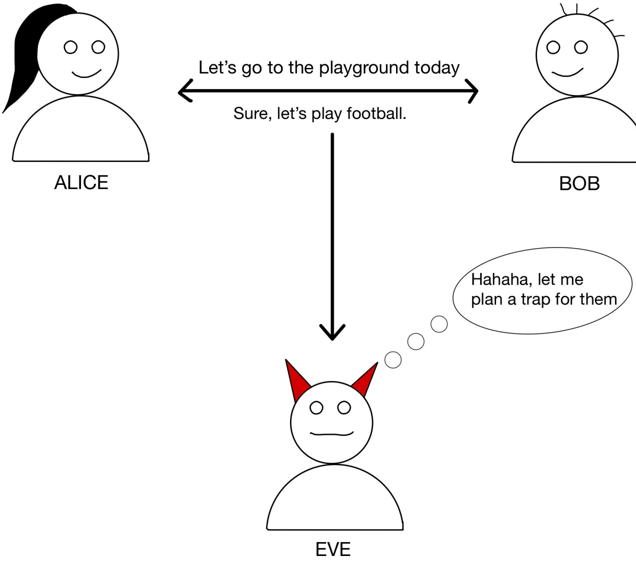
Secure Communication



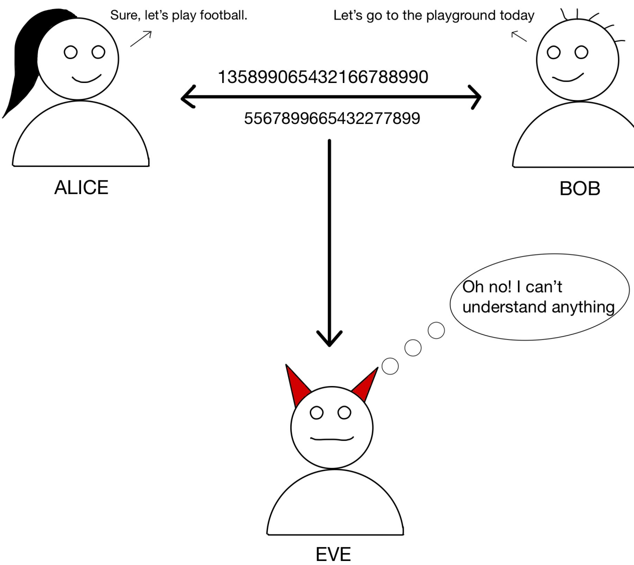
Secure Communication



Secure Communication



Secure Communication



Secure Communication

- Alice wants to send a message m to Bob
- Eve is eavesdropping on every communication
- Alice and Bob don't want Eve to know anything about m
- Alice and Bob can share a *secret* key
- Alice can then send a cipher text

One Time Pad v/s Many Time Pad

- What if Alice wants to send many messages to Bob?

One Time Pad v/s Many Time Pad

- What if Alice wants to send many messages to Bob?
- They can use the same key.

One Time Pad v/s Many Time Pad

- What if Alice wants to send many messages to Bob?
- They can use the same key.
- Will Eve still not be able to find the secret key?

One Time Pad v/s Many Time Pad

- What if Alice wants to send many messages to Bob?
- They can use the same key.
- Will Eve still not be able to find the secret key?
  Crib Dragging

One Time Pad v/s Many Time Pad

- What to do?

One Time Pad v/s Many Time Pad

- What to do?
- Use different key everytime.

One Time Pad v/s Many Time Pad

- What to do?
- Use different key everytime.
- But how to share the secret key?

Number Theory Prelims

Bézout's Identity

Let $a, b \in \mathbb{Z}$ with $(a, b) = d$. Then, $\exists x, y$ such that $ax + by = d$. Moreover, integers of the form $ax + by$ are exactly the multiples of d.

- Extended Euclid's Algorithm can be used to find x, y.

Number Theory Prelims

Fermat's Little Theorem

Let $(a, p) = 1$ where p is a prime. Then

$$a^{p-1} \equiv 1 \mod p$$

Number Theory Prelims

Euler-Fermat's Theorem

Let $(a, n) = 1$. Then

$$a^{\Phi(n)} \equiv 1 \mod n$$

Number Theory Prelims

Chinese Remainder Theorem

Let $m_1, \ldots, m_r \in \mathbb{Z}$ be pairwise co-prime, i.e. $(m_i, m_j) = 1$ if $i \neq j$. Let $a_1, \ldots, a_r \in \mathbb{Z}$. Then the system of congruences,

$$\begin{aligned} x &\equiv a_1 \mod m_1 \\ &\vdots \\ x &\equiv a_r \mod m_r \end{aligned}$$

has a unique solution $\mod m_1 \ldots m_r$.

Garner's Formula

$x = x_2 + (x_1 - x_2)(q^{-1} \mod p)q$

Number Theory Prelims

Fast Modular Exponentiation

```
FastExp (a, b, M)
    ans = FastExp (a, b//2, M)
    ans = (ans · ans)%M
    if b%2
        ans = (ans · (a%M))%M
    return ans
```

RSA Cryptosystem

Brief Overview

- Invented by Rivest, Shamir and Adleman (Turing Award, 2002)
- One of the first public-key encryption based system.
- Most widely used cryptosystem in the world, due to simplicity and reliability.
- Asymmetric Cryptosystem

Brief Overview

- Invented by Rivest, Shamir and Adleman (Turing Award, 2002)
- One of the first public-key encryption based system.
- Most widely used cryptosystem in the world, due to simplicity and reliability.
- **Asymmetric Cryptosystem ?**

Asymmetric vs Symmetric

- Symmetric: Encryption and Decryption keys are same.
- Asymmetric: Different keys are used for Encryption and Decryption.

RSA Protocol

- Bob generates two random keys: $\textsc{Public Key}$ ($\mathbb{E}$) and $\textsc{Private Key}$ ($\mathbb{P}$)
- $\mathbb{E}$ is published so that anyone can access it.
- *Anyone* can encrypt a message for Bob using $\mathbb{E}$, but only Bob can decrypt the message using $\mathbb{P}$.
- The Encryption algorithm is also public, so anyone can try all possible keys to decrypt the message, but it would take hundreds of years for that process with best known algorithms and machines.

What are the keys?

- Bob generates two *Big*, *Random* primes p and q.
- Compute $n = p \cdot q$
- Generate *random* "e" *co-prime* with $(p-1) \cdot (q-1)$
- PUBLIC KEY: $\mathbb{E} = (n, e)$
- PRIVATE KEY: $\mathbb{P} = (p, q)$

Encryption

- Alice: Wants to send message m to Bob.
- m can be encoded as a sequence of bits and converted to an *integer*.
- It is required that m is in the range $[0, n-1]$.
- **Cipher:** $c = m^e \mod n$
- e is called the *public exponent*.
- We can use the *Fast Modular Exponentiation* techniques for this purpose.
- This cipher is sent to Bob.

Decryption

- **Claim:** $\exists\ d$ such that, $c^d \equiv m \mod n$
- d is called the *private exponent*.
- $c^d \equiv (m^e)^d \equiv m^{ed} \mod n$
- We need $m^{ed} \equiv m \mod n$
- By *Euler-Fermat's Theorem*, $m^k \equiv m^{k \mod \phi(n)} \mod n$
- Thus, we finally need,

$$ed \equiv 1 \mod \phi(n) \iff ed \equiv 1 \mod (p-1)(q-1)$$

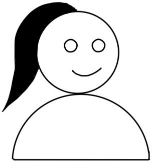
Decryption

- e is co-prime with $(p-1)\cdot(q-1)$.
- d is the modular inverse of e w.r.t. $(p-1)\cdot(q-1)$, and it exists and is unique.
- Thus, d can be calculated immediately when $\mathbb{E}$ is generated using standard techniques (*Extended Euclid's Algorithm)*
- Finally, $m = c^d \mod n$. Fast Modular Exponentiation can be used here too.

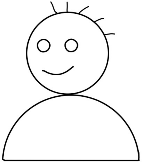
Communication Protocol

- Alice encodes her message m as an integer in $[0, n-1]$.
- She computes the cipher text $c = m^e \mod n$ and sends it to Bob.
- Bob receives the cipher c and decrypts the message $m = c^d \mod n$.

Example



ALICE



BOB

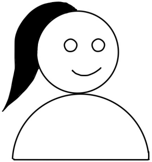
p = 5, q = 11

n = 55, e = 7

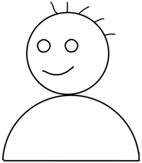
d = 23

Key Generation

Example



ALICE



BOB

$m = 8$

$c = 8^7 \bmod 55$

$= 2$

Encryption

$c = 2$ $\longrightarrow$

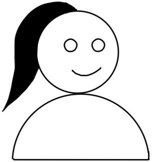
$p = 5,\ q = 11$

$n = 55,\ e = 7$

$d = 23$

Key Generation

Example



ALICE

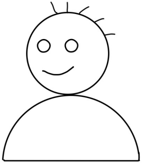
$m = 8$

$c = 8^7 \bmod 55$

$= 2$

Encryption

$c = 2$ $\longrightarrow$



BOB

$p = 5, q = 11$

$n = 55, e = 7$

$d = 23$

Key Generation

$m = 2^{23} \bmod 55$

$= 8$

Decryption

Computational Cost

- Key generation happens occasionally, so not much of an issue.
- **Encryption:**
  - Encoding to integer takes $\mathcal{O}(|m|)$ time.
  - Computing cipher $c = m^e \mod n$ uses the fast modular exponentiation.
  - Time complexity of exponentiation depends on the no. of non-zero bits.
  - Choose integer e such that the number of non-zero bits is less.
  - 3, 17, 65537, etc. are popular choices of primes as e with just two non-zero bits in binary representation.

Computational Cost

- **Decryption:**
  - d can have any number of non-zero bits.
  - Decryption can take more time than Encryption.
  - Taking a small integer as d is risky.
  - Use an alternative method with help of *Chinese Remainder Theorem*.

Efficient Decryption using CRT

- Assume $p > q$. Consider $c^d \equiv m^{ed} \equiv m \mod n$.
- Firstly, $n = p \cdot q$, hence, by *Chinese Remainder Theorem*, it is equivalent to

$$m^{ed} \equiv m \mod p, m^{ed} \equiv m \mod q$$

- By *Fermat's Little Theorem*, $m^{ed} \equiv m^{ed \mod (p-1)} \mod p$
- We need,

$$ed_p \equiv 1 \mod (p-1) \implies d_p \equiv e^{-1} \mod (p-1)$$

- Similarly, $d_q = e^{-1} \mod (q-1)$
- Let $q_{inv} \equiv q^{-1} \mod p$

Efficient Decryption using CRT

- By *Chinese Remainder Theorem*, if $x \equiv x_1 \mod p$ and $x \equiv x_2 \mod q$, then $\exists$ unique $x \in [0, pq)$.
- By *Garner's formula*,

$$x = x_2 + (x_1 - x_2)(q_{inv} \mod p)q$$

Efficient Decryption using CRT

- Let $m_1 = c^{d_p} \mod p, m_2 \equiv c^{d_q} \mod q$.
- Therefore,

$$m = m_2 + (m_1 - m_2)(q_{inv} \mod p)q$$

- This technique speeds up decryption by **4** times.

Is RSA reliable?

- n is known, its factorization $p \cdot q$ is secret.
- RSA relies on the difficulty of the factorization problem.
- If an efficient factorization algorithm appears, then RSA becomes *insecure*.
- It is difficult to find $(p-1) \cdot (q-1)$ given that p and q are secret.
- Otherwise, it would be easy to find d.
- Also, as $(p-1) \cdot (q-1) = \phi(pq) = \phi(n)$, finding $(p-1) \cdot (q-1)$ is equivalent to factorizing $n = pq$.

Is RSA reliable?

- Another viewpoint: *Modular Root Problem*, as we need to find m such that $m^e \equiv c \mod n$ which is equivalent to finding the e^{th} modular root of c.
- This is also a hard problem, but there are some known inefficient algorithms to solve this problem, without the need of factorization.
- If there is a breakthrough in the hard problems like factoring, then RSA is broken. But the converse is an *open problem!*
- Some implementations of RSA are unbreakable, but missing minute details might lead to an unexpected attack.

Attacks on RSA

Types of Attacks

- [3] by *Dan Boneh* is a good survey of attacks on RSA.
- An *attack* is successful if it is able to decrypt the message m in reasonable amount of time.
- The fastest algorithm for factoring is the *General Number Field Sieve* method which has a running time of $\mathcal{O}(exp((c + o(1))n^{\frac{1}{3}} log^{\frac{2}{3}} n))$ for a n-bit number.
- RSA is well-studied, many attacks are known.

Simple Attacks

- **Small Messages:**
  - If the message is small, like “Attack” or “Don’t Attack”.
  - Let’s say encoded as $m = 1$ or $m = 0$.
  - Easy to crack.

Simple Attacks

- **Small Messages:**
  - If the message is small, like "Attack" or "Don't Attack".
  - Let's say encoded as $m = 1$ or $m = 0$.
  - Easy to crack.
  - **Solution:** Pad the messages with random bits.

Simple Attacks

- **Small Messages:**
  - If the message is small, like "Attack" or "Don't Attack".
  - Let's say encoded as $m = 1$ or $m = 0$.
  - Easy to crack.
  - **Solution:** Pad the messages with random bits.
- **Small Prime:**
  - If $p \leqslant 1000000$
  - n can be factorized easily.

Simple Attacks

- **Small Messages:**
  - If the message is small, like "Attack" or "Don't Attack".
  - Let's say encoded as $m = 1$ or $m = 0$.
  - Easy to crack.
  - **Solution:** Pad the messages with random bits.
- **Small Prime:**
  - If $p \leqslant 1000000$
  - n can be factorized easily.
  - **Solution:** Select prime numbers uniformly among very large (2048-bit) numbers.

Small Difference

- What if $|p - q|$ is small?

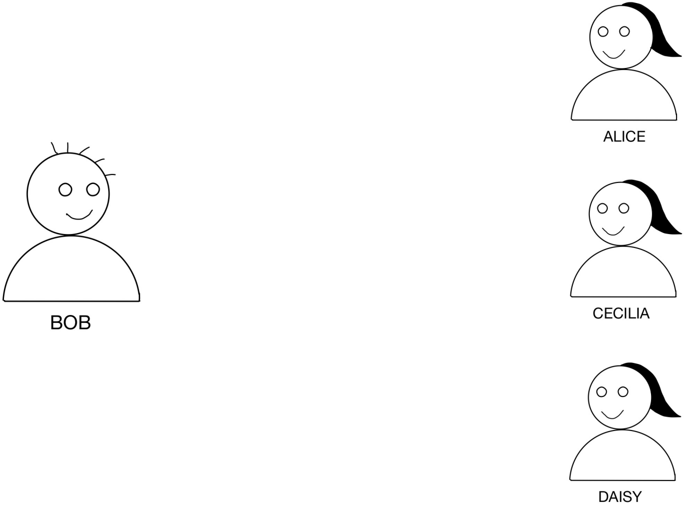
Small Difference

- What if $|p - q|$ is small?
- $n = pq, q < p \implies q < \sqrt{n} < p$
- Let $r = p - q$.
- $\sqrt{n} - q < p - q = r \implies \sqrt{n} - r < q$
- Therefore, $\sqrt{n} - r < q < \sqrt{n}$
- Check for divisors of n in the range $(\sqrt{n} - r, \sqrt{n})$

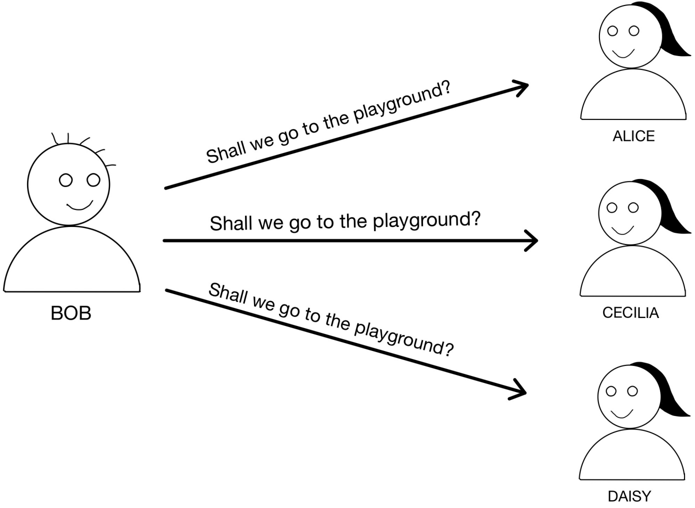
Small Difference

- **Solution:**
- Generate p and q
- If $|p - q|$ is small, generate again
- Repeat until $|p - q|$ is sufficiently large.

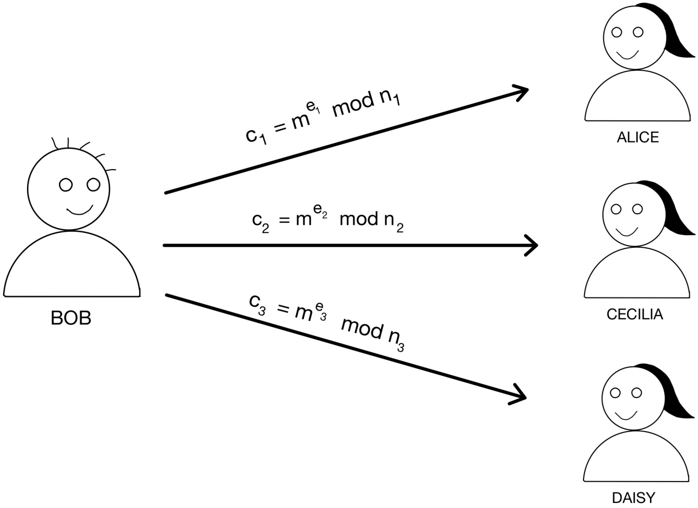
Hastad's Broadcast Attack



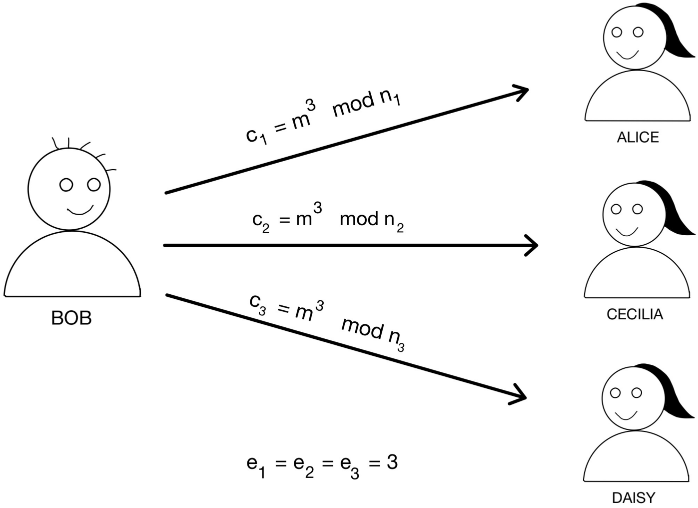
Hastad's Broadcast Attack



Hastad's Broadcast Attack



Hastad's Broadcast Attack



Hastad's Broadcast Attack

- $c_1 \equiv m^3 \mod N_1, c_2 \equiv m^3 \mod N_2, c_3 \equiv m^3 \mod N_3$
- Note, $\text{GCD}(N_i, N_j) = 1$
- By *Chinese Remainder Theorem*, construct c such that $0 \leqslant c < N_1 N_2 N_3$ and $c \equiv c_1 \mod N_1, c \equiv c_2 \mod N_2, c \equiv c_3 \mod N_3$
- $c \equiv m^3 \mod N_1 N_2 N_3$ and $0 \leqslant c, m^3 < N_1 N_2 N_3$
- Therefore, $c = m^3$

Hastad's Broadcast Attack

- Works when $k \geqslant e$.
- Hastad gave a more general theorem.
- **Solution:**
- Use randomized padding.
- Not feasible when e is large.

Other Attacks

- **Insufficient Randomness:** If two public keys have a common prime? *GCD!*
- **Solution:** Random Number Generator must be properly seeded!
- **Low Public Exponent** - Coppersmith's Theorem
- **Franklin-Reiter Related Message Attack**
- **Coppersmith's Short Pad Attack** - Is random padding safe?

Other Attacks

- **Parial Key Exposure Attack** - some bits of d known.
- **One Time Pad v/s Many Time Pad** - Crib Dragging
- **Time taken:** Computing $c^d \mod n$ - If one can send ciphers to a decryption server and sends some response.
- **Power Consumption:** Computing $c^d \mod n$ - Trying to decrypt an encrypted drive.

Conclusion

- RSA is a very powerful yet simple technique.
- The main algorithm is very simple, but care is to be taken during implementation.
- Attacks possible from unexpected angles.

References

- https://www.di-mgt.com.au/rsa_alg.html.
- http://travisdazell.blogspot.com/2012/11/many-time-pad-attack-crib-drag.html.
- Boneh, D.
  Twenty years of attacks on the rsa cryptosystem.
  NOTICES OF THE AMS 46 (02 2002).
- Coursera.
  Number theory and cryptography.
  https://www.coursera.org/learn/number-theory-cryptography/.

Questions?