Lattice Cryptanalysis

Hari

Cryptanalysis

$$\begin{tikzcd}
Alice \arrow[r, "Enc"] & {} \arrow[r, "Enc"] \arrow[d, "Enc"] & Bob \\
& Eve &
\end{tikzcd}$$

Cryptanalysis

- Breaking systems

Cryptanalysis

- Breaking systems
- Partial secret leakage

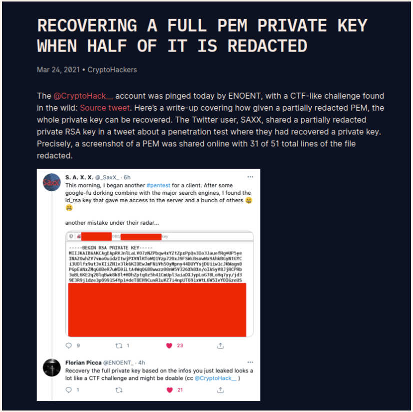
Cryptanalysis

- Breaking systems
- Partial secret leakage
- Poor implementation

Cryptanalysis

- Breaking systems
- Partial secret leakage
- Poor implementation
- Happens a LOT in the real world!

Cryptanalysis

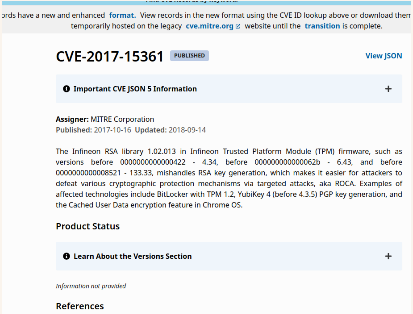


RECOVERING A FULL PEM PRIVATE KEY WHEN HALF OF IT IS REDACTED

Mar 24, 2021 • CryptoHackers

The @CryptoHack__ account was pinged today by ENOENT, with a CTF-like challenge found in the wild: Source tweet. Here's a write-up covering how given a partially redacted PEM, the whole private key can be recovered. The Twitter user, SAXX, shared a partially redacted private RSA key in a tweet about a penetration test where they had recovered a private key. Precisely, a screenshot of a PEM was shared online with 31 of 51 total lines of the file redacted.

Cryptanalysis


ords have a new and enhanced **format.** View records in the new format using the CVE ID lookup above or download them
temporarily hosted on the legacy **cve.mitre.org** website until the **transition** is complete.

CVE-2017-15361 PUBLISHED **View JSON**

Important CVE JSON 5 Information +

Assigner: MITRE Corporation
Published: 2017-10-16 **Updated:** 2018-09-14

The Infineon RSA library 1.02.013 in Infineon Trusted Platform Module (TPM) firmware, such as versions before 0000000000000422 - 4.34, before 000000000000062b - 6.43, and before 0000000000008521 - 133.33, mishandles RSA key generation, which makes it easier for attackers to defeat various cryptographic protection mechanisms via targeted attacks, aka ROCA. Examples of affected technologies include BitLocker with TPM 1.2, YubiKey 4 (before 4.3.5) PGP key generation, and the Cached User Data encryption feature in Chrome OS.

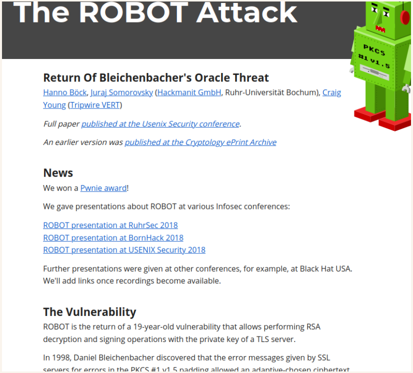
Product Status

Learn About the Versions Section +

Information not provided

References

Cryptanalysis



The ROBOT Attack

PKCS #1 v1.5

Return Of Bleichenbacher's Oracle Threat

Hanno Böck, Juraj Somorovsky (Hackmanit GmbH, Ruhr-Universität Bochum), Craig Young (Tripwire VERT)

Full paper published at the Usenix Security conference.

An earlier version was published at the Cryptology ePrint Archive

News

We won a Pwnie award!

We gave presentations about ROBOT at various Infosec conferences:

ROBOT presentation at RuhrSec 2018
ROBOT presentation at BornHack 2018
ROBOT presentation at USENIX Security 2018

Further presentations were given at other conferences, for example, at Black Hat USA. We'll add links once recordings become available.

The Vulnerability

ROBOT is the return of a 19-year-old vulnerability that allows performing RSA decryption and signing operations with the private key of a TLS server.

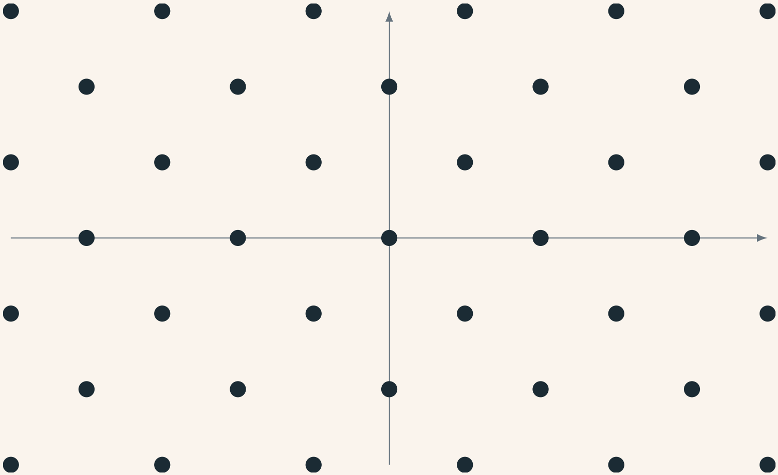
In 1998, Daniel Bleichenbacher discovered that the error messages given by SSL servers for errors in the PKCS #1 v1.5 padding allowed an adaptive-chosen ciphertext

Cryptanalysis

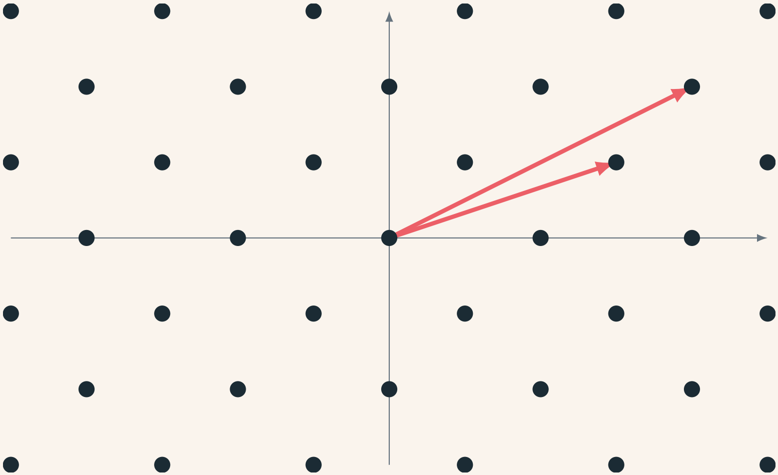
Weak Cryptography Implementations: Fiat-Shamir Attacks On Modern Proofs

Lattices

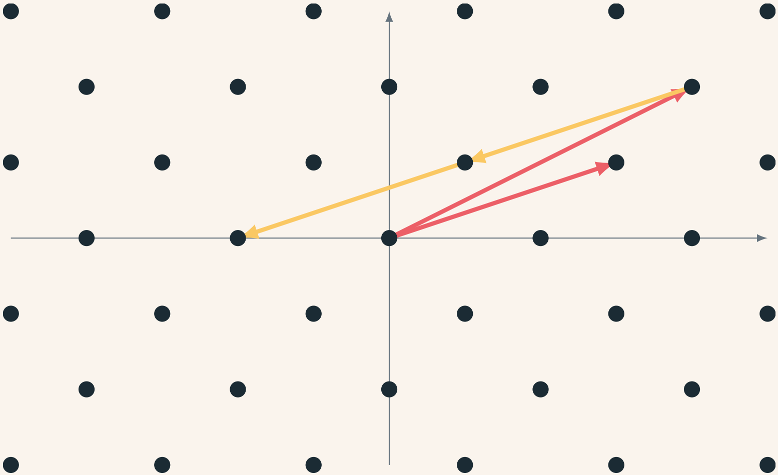
What is a Lattice?



What is a Lattice?



What is a Lattice?



What is a Lattice?

- $a_0\boldsymbol{x} + a_1\boldsymbol{y}$

What is a Lattice?

- $a_0 \boldsymbol{x} + a_1 \boldsymbol{y}$
- **Integer** Linear combination of vectors

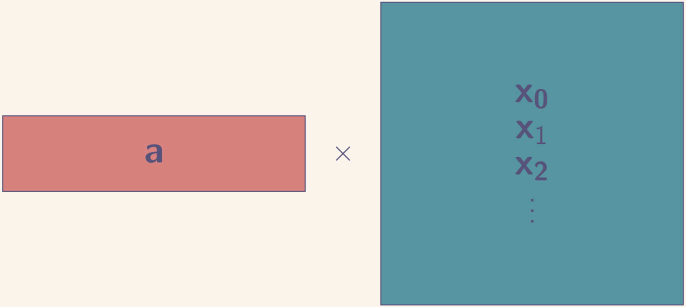
What is a Lattice?

- $a_0 \boldsymbol{x} + a_1 \boldsymbol{y}$
- **Integer** Linear combination of vectors
- Multiple dimensions:
    - $\boldsymbol{x} = [x_0, x_1, x_2, \ldots]$

What is a Lattice?

- $a_0\boldsymbol{x} + a_1\boldsymbol{y}$
- **Integer** Linear combination of vectors
- Multiple dimensions:
  - $\boldsymbol{x} = [x_0, x_1, x_2, \ldots]$
  - $a_0\boldsymbol{x_0} + a_1\boldsymbol{x_1} + \ldots$

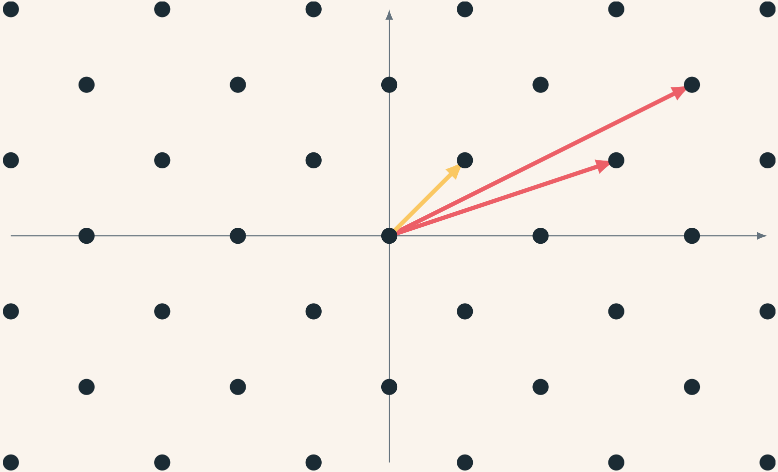
What is a Lattice?



$$a_0\boldsymbol{x_0} + a_1\boldsymbol{x_1} + a_2\boldsymbol{x_3}\ldots$$

$$\boldsymbol{aX}$$

Shortest Vector



Shortest Vector

Definition (Shortest Vector Problem)

Given our $\boldsymbol{X}$, can we find the shortest nonzero point in our lattice?

Shortest Vector

Definition (Shortest Vector Problem)

Given our $\boldsymbol{X}$, can we find the shortest nonzero point in our lattice?

- This problem is hard for computers in high dimensions

Shortest Vector

Definition (Shortest Vector Problem)

Given our $\boldsymbol{X}$, can we find the shortest nonzero point in our lattice?

- This problem is hard for computers in high dimensions
- Can we approximate it?

LLL

- Yes we can!

LLL

- Yes we can!
- We can find a new set of vectors $\boldsymbol{X}'$ which are "somewhat short"

LLL

- Yes we can!
- We can find a new set of vectors $\boldsymbol{X}'$ which are "somewhat short"
- This is known as the Lenstra-Lenstra-Lovasz algorithm

LLL

- Intuition: solve linear equations which are "somewhat short"

LLL

- Intuition: solve linear equations which are "somewhat short"
- Say we have an equation $ax + by = c$ where x and y are small.

LLL

- Intuition: solve linear equations which are "somewhat short"
- Say we have an equation $ax + by = c$ where x and y are small.

$$\begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \\ 0 & 0 & -c \end{bmatrix}$$

LLL

- Intuition: solve linear equations which are "somewhat short"
- Say we have an equation $ax + by = c$ where x and y are small.

$$\begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \\ 0 & 0 & -c \end{bmatrix}$$

Note $[x, y, 0]$ is an integer linear combination

Usages

- Integer Linear programming
- Finding rational approximations to real numbers
- Factorizing rational polynomials
- Solving a modular polynomial for small roots
- Finding minecraft seeds :) (great video on this!)
- **Breaking Cryptosystems**

Case Study 1: Knapsack

Public Key Encryption

- $\mathsf{Gen} \to (pk, sk)$
- $\mathsf{Enc}_{pk}(m) \to c$
- $\mathsf{Dec}_{sk}(c) \to m$

Public Key Encryption

$$
\begin{tikzcd}
\text{Alice} \arrow[r, "m"] & Enc_{pk} \arrow[d, "c"] & \text{Bob's public key} \arrow[l] \\
\text{Bob} & Dec_{sk} \arrow[l, "m"'] & \text{Bob's private key} \arrow[l]
\end{tikzcd}
$$

Public Key Encryption

- Merkle and Hellman proposed a public key system based on "knapsacks"
- Broken by lattices! (with some conditions)

Knapsack Problem

Definition (Knapsack Problem)

Given a set of integers $a_1, a_2, \ldots a_n$, and a target value s, find a subset of those integers that add up to s. Or equivalently, find $e_1, e_2, \ldots e_n$ where $e_i \in \{0, 1\}$ such that

$$\sum_{i=1}^{n} a_i e_i = s$$

Knapsack Problem

Say we have $5, 7, 21, 8, 9, 10$, where the sum has to be equal to 13.

Knapsack Problem

Say we have $5, 7, 21, 8, 9, 10$, where the sum has to be equal to 13.

Solution: $\boxed{5 + 8 = 13}$

Superincreasing Sequences

Definition (Superincreasing Sequence)

We say a set of integers $a_1, a_2, \ldots a_n$ is superincreasing if

$$a_i > \sum_{j=1}^{i-1} a_j$$

Superincreasing Sequences

Definition (Superincreasing Sequence)

We say a set of integers $a_1, a_2, \ldots a_n$ is superincreasing if

$$a_i > \sum_{j=1}^{i-1} a_j$$

- The next number is greater than the sum of the first n numbers

Superincreasing Sequences

Definition (Superincreasing Sequence)

We say a set of integers $a_1, a_2, \ldots a_n$ is superincreasing if

$$a_i > \sum_{j=1}^{i-1} a_j$$

- The next number is greater than the sum of the first n numbers

Superincreasing Sequences

Say we have $1, 5, 8, 20, 35, 80$, where the sum has to be equal to 26.

Superincreasing Sequences

Say we have $1, 5, 8, 20, 35, 80$, where the sum has to be equal to 26.

- Find the largest value less than 26: 20.

Superincreasing Sequences

Say we have $1, 5, 8, 20, 35, 80$, where the sum has to be equal to 26.

- Find the largest value less than 26: 20.
- Subtract it: $26 - 20 = 6$.

Superincreasing Sequences

Say we have $1, 5, 8, 20, 35, 80$, where the sum has to be equal to 26.

- Find the largest value less than 26: 20.
- Subtract it: $26 - 20 = 6$.
- Repeat with 6: $6 - 5 = 1$

Superincreasing Sequences

Say we have $1, 5, 8, 20, 35, 80$, where the sum has to be equal to 26.

- Find the largest value less than 26: 20.
- Subtract it: $26 - 20 = 6$.
- Repeat with 6: $6 - 5 = 1$
- We get $1, 5, 20$

Superincreasing Sequences

Say we have $1, 5, 8, 20, 35, 80$, where the sum has to be equal to 26.

- Find the largest value less than 26: 20.
- Subtract it: $26 - 20 = 6$.
- Repeat with 6: $6 - 5 = 1$
- We get $1, 5, 20$
- Easy to solve Knapsack if superincreasing

Merkle Hellman

- Generate superincreasing sequence $W = (w_1, w_2, \ldots w_n)$

Merkle Hellman

- Generate superincreasing sequence $W = (w_1, w_2, \ldots w_n)$
- Choose a prime q greater than the sum of W

Merkle Hellman

- Generate superincreasing sequence $W = (w_1, w_2, \ldots w_n)$
- Choose a prime q greater than the sum of W
- Choose r and s such that $r \cdot s = 1 \mod q$

Merkle Hellman

- Generate superincreasing sequence $W = (w_1, w_2, \ldots w_n)$
- Choose a prime q greater than the sum of W
- Choose r and s such that $r \cdot s = 1 \mod q$
- Calculate new sequence $B = (b_i = r \cdot w_i \mod q)$

Merkle Hellman

- Generate superincreasing sequence $W = (w_1, w_2, \ldots w_n)$
- Choose a prime q greater than the sum of W
- Choose r and s such that $r \cdot s = 1 \mod q$
- Calculate new sequence $B = (b_i = r \cdot w_i \mod q)$
- Public: B
- Private: W, r, s, q

Merkle Hellman

- Encryption of message:

Merkle Hellman

- Encryption of message:
- Convert message into bits $m = m_1 m_2 m_3 \ldots$

Merkle Hellman

- Encryption of message:
- Convert message into bits $m = m_1 m_2 m_3 \ldots$
- Calculate ciphertext as the knapsack sum of B and m:

$$c = \sum_{i=1}^{n} b_i \cdot m_i$$

Merkle Hellman

- Decryption of ciphertext:

Merkle Hellman

- Decryption of ciphertext:
- Remember that c is

$$c = \sum_{i=1}^{n} b_i \cdot m_i = \sum_{i=1}^{n} rw_i \cdot m_i$$

Merkle Hellman

- Decryption of ciphertext:
- Remember that c is

$$c = \sum_{i=1}^{n} b_i \cdot m_i = \sum_{i=1}^{n} rw_i \cdot m_i$$

- Calculate $c' = sc \mod q$

$$c' = s \sum_{i=1}^{n} rw_i \cdot m_i = sr \sum_{i=1}^{n} w_i \cdot m_i = \sum_{i=1}^{n} w_i \cdot m_i$$

Merkle Hellman

- Decryption of ciphertext:
- Remember that c is

$$c = \sum_{i=1}^{n} b_i \cdot m_i = \sum_{i=1}^{n} rw_i \cdot m_i$$

- Calculate $c' = sc \mod q$

$$c' = s\sum_{i=1}^{n} rw_i \cdot m_i = sr\sum_{i=1}^{n} w_i \cdot m_i = \sum_{i=1}^{n} w_i \cdot m_i$$

- Now we can solve, because we know the superincreasing sequence W.

Merkle Hellman

- Decryption of ciphertext:
- Remember that c is

$$c = \sum_{i=1}^{n} b_i \cdot m_i = \sum_{i=1}^{n} rw_i \cdot m_i$$

- Calculate $c' = sc \mod q$

$$c' = s\sum_{i=1}^{n} rw_i \cdot m_i = sr\sum_{i=1}^{n} w_i \cdot m_i = \sum_{i=1}^{n} w_i \cdot m_i$$

- Now we can solve, because we know the superincreasing sequence W.
- Note: If we can solve the knapsack problem, we don't need the private key!

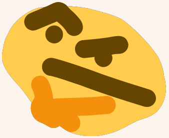
Knapsack Problem

Definition (Knapsack Problem)

Given a set of integers $a_1, a_2, \ldots a_n$, and a target value s, find a subset of those integers that add up to s. Or equivalently, find $e_1, e_2, \ldots e_n$ where $e_i \in \{0, 1\}$ such that

$$\sum_{i=1}^{n} a_i e_i = s$$

Hmm looks like a short linear combination to me



Solving Knapsack*

$$\begin{bmatrix} 1 & 0 & 0 & 0 & b_1 \\ 0 & 1 & 0 & 0 & b_2 \\ 0 & 0 & 1 & 0 & b_3 \\ 0 & 0 & 0 & 1 & b_4 \\ 0 & 0 & 0 & 0 & -c \end{bmatrix}$$

Short vector of all 0 and 1 : $[m_1, m_2, m_3, m_4, 0]$

Demo

Demo!

Case Study 2: Secret Sharing

Secret Sharing

Definition ((t, n) secret sharing scheme)

A (t, n) secret sharing scheme for secret s is defined to be

- If any t people get together, they can learn the secret
- If any $t - 1$ get together, they learn nothing

Secret Sharing

Definition ((t, n) secret sharing scheme)

A (t, n) secret sharing scheme for secret s is defined to be

- If any t people get together, they can learn the secret
- If any $t - 1$ get together, they learn nothing

- Split the secret into smaller "shares" for individual people

Secret Sharing

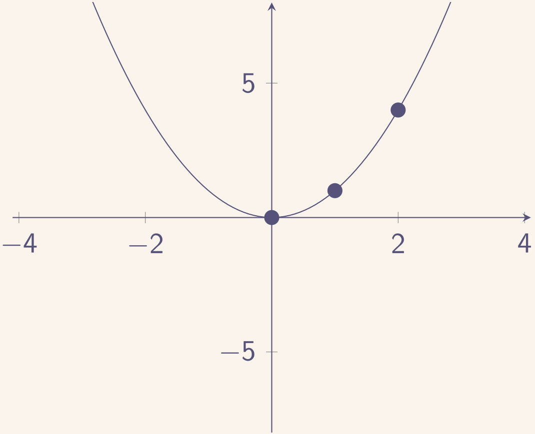
Definition ((t, n) secret sharing scheme)

A (t, n) secret sharing scheme for secret s is defined to be

- If any t people get together, they can learn the secret
- If any $t - 1$ get together, they learn nothing

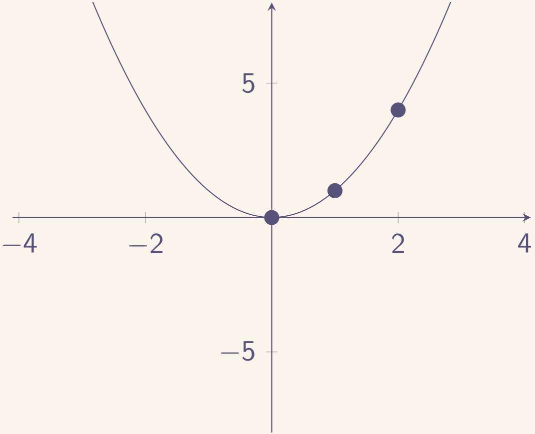
- Split the secret into smaller "shares" for individual people
- Useful for storing secrets (for example blockchain wallets)
- Password managers, company key sharing

Polynomials



- x^2: degree 2

Polynomials



- x^2: degree 2
- Degree t polynomial uniquely defined by $t+1$ points

Shamir's Secret Sharing

- Let $f(x) = a_{t-1}x^{t-1} + a_{t-2}x^{t-2} \ldots + a_2x^2 + a_1x + s \mod q$

Shamir's Secret Sharing

- Let $f(x) = a_{t-1}x^{t-1} + a_{t-2}x^{t-2} \ldots + a_2x^2 + a_1x + s \mod q$
- Give n people each a point $(x_i, f(x_i))$

Shamir's Secret Sharing

- Let $f(x) = a_{t-1}x^{t-1} + a_{t-2}x^{t-2} \ldots + a_2x^2 + a_1x + s \mod q$
- Give n people each a point $(x_i, f(x_i))$
- Any t people can reconstruct the polynomial and find s

Poor implementation

- **what happens if we forget the mod q?**

Poor implementation

- **what happens if we forget the mod q?**
- Say we have $n = 5, t = 3$
- We only have two shares $(x_1, f(x_1)), (x_2, f(x_2))$

Poor implementation

- **what happens if we forget the mod q?**
- Say we have $n = 5, t = 3$
- We only have two shares $(x_1, f(x_1)), (x_2, f(x_2))$
- Recall $f(x) = ax^2 + bx + s$

Poor implementation

- **what happens if we forget the mod q?**
- Say we have $n = 5, t = 3$
- We only have two shares $(x_1, f(x_1)), (x_2, f(x_2))$
- Recall $f(x) = ax^2 + bx + s$
- Two linear equations: $ax_i^2 + bx_i + s$

Poor implementation

$$\begin{bmatrix} 1 & 0 & 0 & kx_1^2 & kx_2^2 \\ 0 & 1 & 0 & kx_1 & kx_2 \\ 0 & 0 & 1 & k & k \\ 0 & 0 & 0 & -kf(x_1) & -kf(x_2) \end{bmatrix}$$

- If k is very large: short vector $[a, b, s, 0, 0]$

Poor implementation

$$\begin{bmatrix} 1 & 0 & 0 & kx_1^2 & kx_2^2 \\ 0 & 1 & 0 & kx_1 & kx_2 \\ 0 & 0 & 1 & k & k \\ 0 & 0 & 0 & -kf(x_1) & -kf(x_2) \end{bmatrix}$$

- If k is very large: short vector $[a, b, s, 0, 0]$
- The more shares you have, the more likely to be correct

Poor implementation

$$\begin{bmatrix} 1 & 0 & 0 & kx_1^2 & kx_2^2 \\ 0 & 1 & 0 & kx_1 & kx_2 \\ 0 & 0 & 1 & k & k \\ 0 & 0 & 0 & -kf(x_1) & -kf(x_2) \end{bmatrix}$$

- If k is very large: short vector $[a, b, s, 0, 0]$
- The more shares you have, the more likely to be correct

Demo

Demo!

More Fun Things

- Partial RSA information
- Factorize a number $n = pq$ given the "top part" of p

More Fun Things

- Partial RSA information
- Factorize a number $n = pq$ given the "top part" of p
- Used to find minecraft seeds
- Java random uses a linear random number generator
- Solve for specific situations: 12 eyes, etc

Final Thoughts

- What lattices are
- Attacked a public key cryptosystem
- Attacked a poor implementation