Improved Power Analysis Attacks on Falcon

EUROCRYPT 2023

Shiduo Zhang, Xiuhan Lin, Yang Yu, Weijia Wang

清華大學 Tsinghua University

山东大学 SHANDONG UNIVERSITY

Overview

In this work, we develop several key recovery attacks exploiting **power leakage** on Falcon.

- a new effective key recovery using the half Gaussian leakage within the base sampler.
- the first side-channel analysis on Falcon taking the sign leakage into account.

Outline

1. Background
2. The half Gaussian leakage and the sign leakage
3. Exploiting the half Gaussian leakage
4. Exploiting the sign leakage

Background

Falcon

Falcon is one of the three post quantum digital signatures to be standardized by NIST.

Falcon has a good performance especially it has the smallest bandwidth (public key size plus signature size) among the selected NIST signatures.

Falcon is a lattice-based hash-and-sign signature scheme.

Hash-and-sign paradigm

Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

Hash-and-sign paradigm

Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

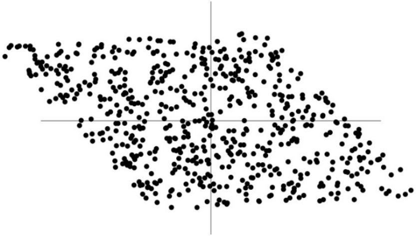
GGH, NTRUSign used the deterministic Babai's algorithm to find the close vectors.

Hash-and-sign paradigm

Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign → GPV → Falcon

GGH, NTRUSign used the deterministic Babai's algorithm to find the close vectors.

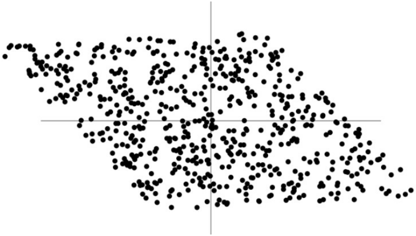


Hash-and-sign paradigm

Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

GGH, NTRUSign used the deterministic Babai's algorithm to find the close vectors. **Insecure !**



Hash-and-sign paradigm

Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

GGH, NTRUSign used the deterministic Babai's algorithm to find the close vectors. **Insecure !**

[GPV08] proposed a provably secure hash-and-sign framework[1].

- rounding based on random Gaussian sampling
- distribution of signatures is provably independent of the secret key

[1] [GPV08] :Trapdoors for Hard Lattices and New Cryptographic Constructions, Gentry, Peikert, Vaikuntanathan

Hash-and-sign paradigm

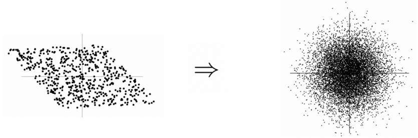
Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

GGH, NTRUSign used the deterministic Babai's algorithm to find the close vectors. **Insecure !**

[GPV08] proposed a provably secure hash-and-sign framework[1].

- rounding based on random Gaussian sampling
- distribution of signatures is provably independent of the secret key



[1] [GPV08] :Trapdoors for Hard Lattices and New Cryptographic Constructions, Gentry, Peikert, Vaikuntanathan

Hash-and-sign paradigm

Hash-and-sign

- signing: finding close vectors
- GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

GGH, NTRUSign used the deterministic Babai's algorithm to find the close vectors. **Insecure !**

[GPV08] proposed a provably secure hash-and-sign framework[1].

- rounding based on random Gaussian sampling
- distribution of signatures is provably independent of the secret key

Falcon is an efficient instantiation of the GPV framework by using optimal NTRU trapdoor.

[1] [GPV08] :Trapdoors for Hard Lattices and New Cryptographic Constructions. Gentry, Peikert, Vaikuntanathan.

Side-channel analysis

With PQC standardization and migration underway, security should be considered from both **algorithmic** and **implementation** aspects.

The implementation security of Falcon is intricate

- signing relies on complicated lattice Gaussian sampling
- secret key is used in a rather opaque way

Side-channel analysis

With PQC standardization and migration underway, security should be considered from both **algorithmic** and **implementation** aspects.

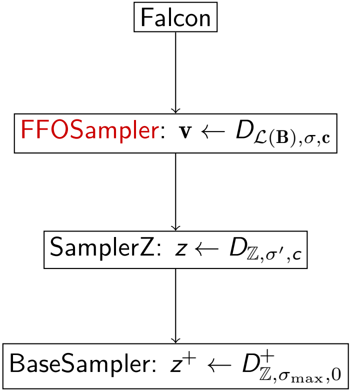
The implementation security of Falcon is intricate

- signing relies on complicated lattice Gaussian sampling
- secret key is used in a rather opaque way

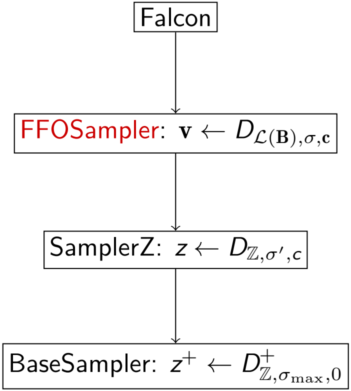
We need to understand the connection between leakage and secret key itself.

Gaussian Samplers of Falcon

Sampler


Falcon
FFOSampler: $\mathbf{v} \leftarrow D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}}$
SamplerZ: $z \leftarrow D_{\mathbb{Z},\sigma',c}$
BaseSampler: $z^{+} \leftarrow D^{+}_{\mathbb{Z},\sigma_{\max},0}$

Sampler



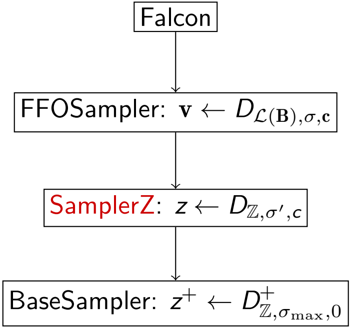
The KGPV sampler

Input: a basis $\mathbf{B} = (\mathbf{b}_0, \cdots, \mathbf{b}_{n-1})$, a center $\mathbf{c}$ and $\sigma \geq \|\mathbf{B}\|_{GS} \cdot \eta_\epsilon(\mathbb{Z})$

Output: a lattice point $\mathbf{v}$ following a distribution close to $D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}}$.

$\mathbf{v} \leftarrow \mathbf{0}, \mathbf{c}' \leftarrow \mathbf{c}$
for $i = n-1, \cdots, 0$ **do**
  $\sigma_i = \sigma / \|\widetilde{\mathbf{b}}_i\|$
  $c_i'' = \langle \mathbf{c}', \widetilde{\mathbf{b}}_i \rangle / \|\widetilde{\mathbf{b}}_i\|^2$
  $z_i \leftarrow$ SamplerZ$(\sigma_i, c_i'' - \lfloor c_i'' \rceil) + \lfloor c_i'' \rceil$
  $\mathbf{c}' \leftarrow \mathbf{c}' - z_i \mathbf{b}_i, \mathbf{v} \leftarrow \mathbf{v} + z_i \mathbf{b}_i$
end for
return $\mathbf{v}$

Integer Gaussian sampler of Falcon

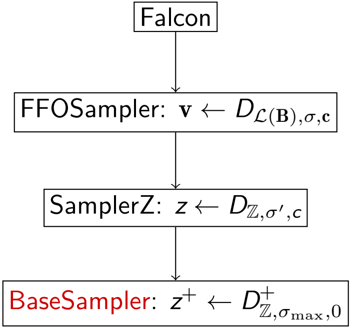


SamplerZ(σ, c)

Input: $c \in [0,1)$ and $\sigma \in (\sigma_{min}, \sigma_{max})$.
Output: $z \in \mathbb{Z}$ following $D_{\mathbb{Z},\sigma,c}$.

1: $z^+ \leftarrow$ BaseSampler()
2: $b \leftarrow U(\{0,1\})$
3: $z \leftarrow b + (2b-1)z^+$
4: $x \leftarrow -\frac{(z-c)^2}{2\sigma^2} + \frac{(z^+)^2}{2\sigma_{max}^2}$
5: return z with probability $\frac{\sigma_{min}}{\sigma} \cdot \exp(x)$, otherwise restart;

Integer Gaussian sampler of Falcon



BaseSampler()

Output: $z^+ \sim D^+_{\mathbb{Z},\sigma_{\max},0}$.

$u \xleftarrow{\$} \{0,1\}^{72}$
$z^+ \leftarrow 0$
for $i = 0 \cdots 17$ **do**
    $z^+ \leftarrow z^+ + [\![u < RCDT[i]]\!]$
end for
return z^+

The half Gaussian leakage and the sign leakage

Half Gaussian leakages

BaseSampler()

Output: $z^+ \sim D^+_{\mathbb{Z},\sigma_{\max},0}$.

```
u ←$ {0,1}^72
z+ ← 0
for i = 0 ··· 17 do
    z+ ← z+ + [[u < RCDT[i]]]
end for
return z+
```

Half Gaussian leakages

BaseSampler()

Output: $z^+ \sim D^+_{\mathbb{Z},\sigma_{\max},0}$.

1: $u \xleftarrow{\$} \{0,1\}^{72}$
2: $z^+ \leftarrow 0$
3: **for** $i = 0 \cdots 17$ **do**
4: $\quad z^+ \leftarrow z^+ + [\![u < RCDT[i]]\!]$
5: **end for**
6: return z^+

Half Gaussian leakage

One can classify if $z^+ = 0$ or not through the power consumption of the comparison $[\![u < RCDT[i]]\!]$

Sign leakages

SamplerZ(σ, c)

Input: $c \in [0, 1)$ and $\sigma \in (\sigma_{min}, \sigma_{max})$.
Output: $z \in \mathbb{Z}$ following $D_{\mathbb{Z},\sigma,c}$.

1: $z^+ \leftarrow$ BaseSampler()
2: $b \leftarrow U(\{0, 1\})$
3: $z \leftarrow b + (2b - 1)z^+$
4: $x \leftarrow -\frac{(z-c)^2}{2\sigma^2} + \frac{(z^+)^2}{2\sigma_{max}^2}$
5: return z with probability $\frac{\sigma_{min}}{\sigma} \cdot \exp(x)$, otherwise restart;

Sign leakages

SamplerZ(σ, c)

Input: $c \in [0,1)$ and $\sigma \in (\sigma_{min}, \sigma_{max})$.
Output: $z \in \mathbb{Z}$ following $D_{\mathbb{Z},\sigma,c}$.

1: $z^+ \leftarrow$ BaseSampler()
2: $b \leftarrow U(\{0,1\})$
3: $z \leftarrow b + (2b-1)z^+$
4: $x \leftarrow -\frac{(z-c)^2}{2\sigma^2} + \frac{(z^+)^2}{2\sigma_{max}^2}$
5: return z with probability $\frac{\sigma_{min}}{\sigma} \cdot \exp(x)$, otherwise restart;

Sign leakage

One can determine b through the power of $[\![z \leftarrow b + (2b-1)z^+]\!]$ and $[\![x \leftarrow -\frac{(z-c)^2}{2\sigma'^2} + \frac{(z^+)^2}{2\sigma_{max}^2}]\!]$

Exploiting the half Gaussian leakage

Parallelepiped-learning strikes again

In [GMRR22], Guerreau et al. proposed a key recovery attack exploiting the half Gaussian leakage.[2]

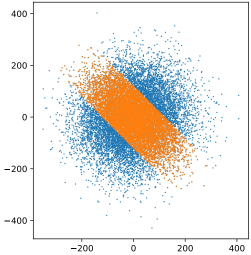
[2] The Hidden Parallelepiped Is Back Again: Power Analysis Attacks on Falcon. Guerreau, Martinelli, Ricosset, Rossi.

Parallelepiped-learning strikes again

In [GMRR22], Guerreau et al. proposed a key recovery attack exploiting the half Gaussian leakage.[2]

Fact

When $z_0^+ = 0$, the signature $\mathbf{s} = \sum_{i=0}^{2n-1} y_i \cdot \tilde{\mathbf{b}}_i$ with $y_0 \in [-1, 1]$.



The attack of [GMRR22]

They reused the parallelepiped-learning technique to recover the key

The attack is **rather expensive**.

- 10^7 traces for direct recovery
- 10^6 traces and 1000h of computation

[2] The Hidden Parallelepiped Is Back Again: Power Analysis Attacks on Falcon. Guerreau, Martinelli, Ricosset, Rossi.

Our improved key recovery

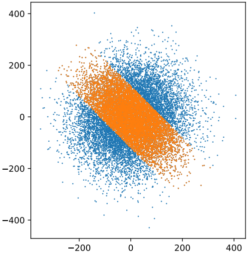
Learning Slice Problem $\mathsf{LSP}_{b,\sigma,N}$

Given $\mathbf{b} \in \mathbb{R}^n$, let $\mathcal{S}_{\mathbf{b}}(b) = \{\mathbf{v} : |\langle \mathbf{v}, \mathbf{b} \rangle| \le b\}$. Let D_s be the conditional distribution of $\mathbf{z} \sim (\mathcal{N}(0,\sigma^2))^n$ given $\mathbf{z} \in \mathcal{S}_{\mathbf{b}}(b)$. Given N independent samples drawn from D_s, find an approximation of $\pm\mathbf{b}$.

The geometric intuition: the projection of signatures in the slice on $\mathbf{b}_0$ tends to be **unusually short**.

Our LSP algorithm

1. learning the direction of $\mathbf{b}_0$
2. estimating $\|\mathbf{b}_0\|$



Step 1: Learning the slice direction

Let $\mathbf{B} = (\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_{n-1})$ and $\mathbf{D} = (\mathbf{d}_0, \cdots, \mathbf{d}_{n-1})$ with $\mathbf{d}_i = \widetilde{\mathbf{b}}_i / \|\widetilde{\mathbf{b}}_i\|$.

For $\mathbf{s} = \sum_i y_i \mathbf{d}_i \sim (\mathcal{N}(0, \sigma^2))^n$, $y_i \sim \mathcal{N}(0, \sigma^2)$ and $\mathbf{Cov}[\mathbf{s}] = \sigma^2 I$.

When $\mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)$, the variance of y_0 is $\sigma'^2 < \sigma^2$ and thus

$$\mathbf{Cov}[\mathbf{s} | \mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)] = \mathbf{D} \cdot \begin{pmatrix} \sigma'^2 & \\ & \sigma^2 I \end{pmatrix} \cdot \mathbf{D}^t.$$

Step 1: Learning the slice direction

Let $\mathbf{B} = (\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_{n-1})$ and $\mathbf{D} = (\mathbf{d}_0, \cdots, \mathbf{d}_{n-1})$ with $\mathbf{d}_i = \widetilde{\mathbf{b}}_i / \|\widetilde{\mathbf{b}}_i\|$.

For $\mathbf{s} = \sum_i y_i \mathbf{d}_i \sim (\mathcal{N}(0, \sigma^2))^n$, $y_i \sim \mathcal{N}(0, \sigma^2)$ and $\mathbf{Cov}[\mathbf{s}] = \sigma^2 I$.

When $\mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)$, the variance of y_0 is $\sigma'^2 < \sigma^2$ and thus

$$\mathbf{Cov}[\mathbf{s} | \mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)] = \mathbf{D} \cdot \begin{pmatrix} \sigma'^2 & \\ & \sigma^2 I \end{pmatrix} \cdot \mathbf{D}^t.$$

Fact

The smallest eigenvalue σ' is unique and its eigenvector is in the same direction as $\mathbf{b}_0$.

Step 1: Learning the slice direction

Let $\mathbf{B} = (\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_{n-1})$ and $\mathbf{D} = (\mathbf{d}_0, \cdots, \mathbf{d}_{n-1})$ with $\mathbf{d}_i = \widetilde{\mathbf{b}}_i / \|\widetilde{\mathbf{b}}_i\|$.

For $\mathbf{s} = \sum_i y_i \mathbf{d}_i \sim (\mathcal{N}(0, \sigma^2))^n$, $y_i \sim \mathcal{N}(0, \sigma^2)$ and $\mathbf{Cov}[\mathbf{s}] = \sigma^2 I$.

When $\mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)$, the variance of y_0 is $\sigma'^2 < \sigma^2$ and thus

$$\mathbf{Cov}[\mathbf{s} | \mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)] = \mathbf{D} \cdot \begin{pmatrix} \sigma'^2 & \\ & \sigma^2 I \end{pmatrix} \cdot \mathbf{D}^t.$$

Fact

The smallest eigenvalue σ' is unique and its eigenvector is in the same direction as $\mathbf{b}_0$.

This allows us to recover the direction through **spectral decomposition**!

Step 1: Learning the slice direction

Let $\mathbf{B} = (\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_{n-1})$ and $\mathbf{D} = (\mathbf{d}_0, \cdots, \mathbf{d}_{n-1})$ with $\mathbf{d}_i = \widetilde{\mathbf{b}}_i / \|\widetilde{\mathbf{b}}_i\|$.

For $\mathbf{s} = \sum_i y_i \mathbf{d}_i \sim (\mathcal{N}(0, \sigma^2))^n$, $y_i \sim \mathcal{N}(0, \sigma^2)$ and $\mathbf{Cov}[\mathbf{s}] = \sigma^2 I$.

When $\mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)$, the variance of y_0 is $\sigma'^2 < \sigma^2$ and thus

$$\mathbf{Cov}[\mathbf{s} | \mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)] = \mathbf{D} \cdot \begin{pmatrix} \sigma'^2 & \\ & \sigma^2 I \end{pmatrix} \cdot \mathbf{D}^t.$$

Fact

The smallest eigenvalue σ' is unique and its eigenvector is in the same direction as $\mathbf{b}_0$.

This allows us to recover the direction through **spectral decomposition**!

This analysis can be understood as principal component analysis rather than independent component analysis.

Step 2: Learning the norm

The covariance $\mathbf{Cov}[\mathbf{s}|\mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)]$ also leaks the information of $\|\mathbf{b}_0\|$:

$$\sigma'^2 = \frac{\int_{-b'}^{b'} x^2 \exp(-\frac{x^2}{2\sigma^2}) dx}{\int_{-b'}^{b'} \exp(-\frac{x^2}{2\sigma^2}) dx} \quad \text{where} \quad b' = \frac{b}{\|\mathbf{b}_0\|}.$$

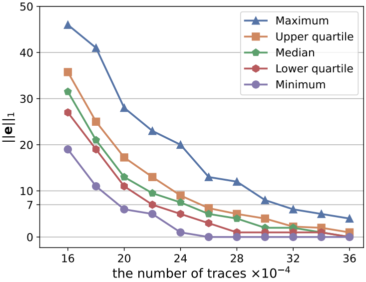
Step 2: Learning the norm

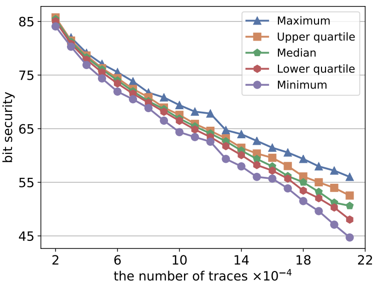
The covariance $\mathbf{Cov}[\mathbf{s}|\mathbf{s} \in \mathcal{S}_{\mathbf{b}_0}(b)]$ also leaks the information of $\|\mathbf{b}_0\|$:

$$\sigma'^2 = \frac{\int_{-b'}^{b'} x^2 \exp(-\frac{x^2}{2\sigma^2}) dx}{\int_{-b'}^{b'} \exp(-\frac{x^2}{2\sigma^2}) dx} \quad \text{where} \quad b' = \frac{b}{\|\mathbf{b}_0\|}.$$

This allows to numerically estimate $\|\mathbf{b}_0\|$!

Experimental results





Our attack is much more efficient compared with [GMRR22][3]!

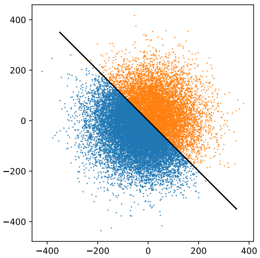
- direct recovery: 10^7 traces $\rightarrow 3.6 \times 10^5$ traces
- 10^6 traces + 1000h $\rightarrow 2.2 \times 10^5$ traces + 0.5h of computation

[3] The Hidden Parallelepiped Is Back Again: Power Analysis Attacks on Falcon. Guerreau, Martinelli, Ricosset, Rossi.

Exploiting the sign leakage

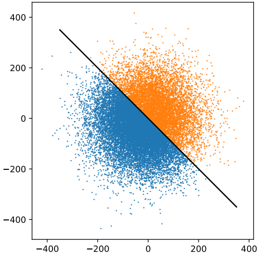
Learning the halfspace

The sign leakage allows to determine whether a signature $\mathbf{s}$ is in the halfspace $\mathcal{H}^+ = \{\mathbf{v} : \langle \mathbf{v}, \mathbf{b}_0 \rangle \geq 0\}$ or $\mathcal{H}^- = \{\mathbf{v} : \langle \mathbf{v}, \mathbf{b}_0 \rangle < 0\}$



Learning the halfspace

The sign leakage allows to determine whether a signature $\mathbf{s}$ is in the halfspace $\mathcal{H}^+ = \{\mathbf{v} : \langle \mathbf{v}, \mathbf{b}_0 \rangle \geq 0\}$ or $\mathcal{H}^- = \{\mathbf{v} : \langle \mathbf{v}, \mathbf{b}_0 \rangle < 0\}$



Learning Halfspace Problem $\mathrm{LHP}_{\sigma,N}$

Given $\mathbf{b} \in \mathbb{R}^n$, let $\mathcal{H}_{\mathbf{b}}^+ = \{\mathbf{v} : \langle \mathbf{v}, \mathbf{b} \rangle \geq 0\}$. Let D_h be the conditional distribution of $\mathbf{z} \sim (\mathcal{N}(0, \sigma^2))^n$ given $\mathbf{z} \in \mathcal{H}_{\mathbf{b}}^+$. Given N independent samples drawn from D_h, find an approximate direction of $\pm\mathbf{b}$.

Our LHP algorithm

At a high level, our algorithm can be seen as the reduction:

$$\mathsf{LHP}_{\sigma,N} \to \mathsf{LSP}_{b,\sigma,N'}.$$

Our LHP algorithm

At a high level, our algorithm can be seen as the reduction:

$$\mathsf{LHP}_{\sigma,N} \to \mathsf{LSP}_{b,\sigma,N'}.$$

Our LHP algorithm

1. learning a relatively rough direction $\mathbf{v}$ of $\mathbf{b}_0$ from samples in $\mathcal{H}^+_{\mathbf{b}_0}$
2. filtering out those samples in $\mathcal{S}_{\mathbf{v}}(b)$ using $\mathbf{v}$
3. learning the direction of $\mathbf{b}_0$ from the filtered samples in $\mathcal{S}_{\mathbf{v}}(b)$

Step 1: Learning a rough direction

The coefficient of $\mathbf{d}_0$ is half Gaussian, while others are full Gaussian.
$\Rightarrow$ The direction can be learned through spectral decomposition as well!

Step 1: Learning a rough direction

The coefficient of $\mathbf{d}_0$ is half Gaussian, while others are full Gaussian.
$\Rightarrow$ The direction can be learned through spectral decomposition as well!

Since the gap between the smallest eigenvalue and others now increases, an accurate approximation requires more samples.
$\Rightarrow$ We learn a relatively rough direction instead of a very accurate one

Step 1: Learning a rough direction

The coefficient of $\mathbf{d}_0$ is half Gaussian, while others are full Gaussian.
$\Rightarrow$ The direction can be learned through spectral decomposition as well!

Since the gap between the smallest eigenvalue and others now increases, an accurate approximation requires more samples.
$\Rightarrow$ We learn a relatively rough direction instead of a very accurate one

One can also learn the direction through the expectation of samples, but the expectation does not seem to improve the attack

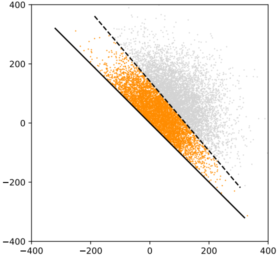
Step 2: Filtering out a slice

To refine the accuracy, we attempt to amplify the condition number.

Step 2: Filtering out a slice

To refine the accuracy, we attempt to amplify the condition number.

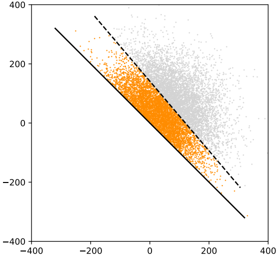
We propose to use the rough direction $\mathbf{v}$ to classify all samples into two sets $\mathcal{S} = \{\mathbf{s} \mid |\langle \mathbf{s}, \mathbf{v} \rangle| \leq b\}$ and $\mathcal{C} = \{\mathbf{s} \mid |\langle \mathbf{s}, \mathbf{v} \rangle| > b\}$



Step 2: Filtering out a slice

To refine the accuracy, we attempt to amplify the condition number.

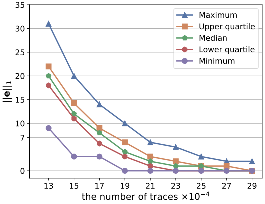
We propose to use the rough direction $\mathbf{v}$ to classify all samples into two sets $\mathcal{S} = \{\mathbf{s} \mid |\langle \mathbf{s}, \mathbf{v} \rangle| \leq b\}$ and $\mathcal{C} = \{\mathbf{s} \mid |\langle \mathbf{s}, \mathbf{v} \rangle| > b\}$

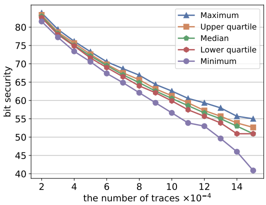


Applying our LSP algorithm, we obtain a more accurate direction!

Experimental results

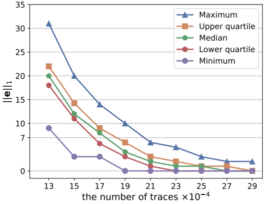
The attack is more efficient than the one using half Gaussian leakages.

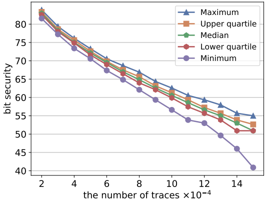




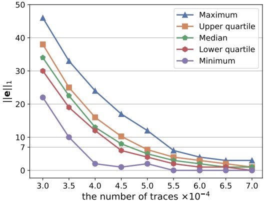
Experimental results

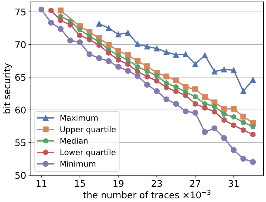
The attack is more efficient than the one using half Gaussian leakages.





The attack can be more efficient by using both two leakages!





A practical countermeasure

SamplerZ(σ, c)

Input: $c \in [0, 1)$ and $\sigma \in (\sigma_{min}, \sigma_{max})$.
Output: $z \sim D_{\mathbb{Z},\sigma,c}$.

1: $z^+ \leftarrow$ BaseSampler()
2: $\mathbf{b \leftarrow U(\{0, 1\})}$
3: $\mathbf{z \leftarrow b + (2b - 1)z^+}$
4: $\mathbf{x \leftarrow -\frac{(z-c)^2}{2\sigma^2} + \frac{(z^+)^2}{2\sigma_{max}^2}}$
5: return z with probability $\frac{\sigma_{min}}{\sigma} \cdot \exp(x)$, otherwise restart;

A practical countermeasure

SamplerZ(σ, c)

Input: $c \in [0, 1)$ and $\sigma \in (\sigma_{min}, \sigma_{max})$.
Output: $z \sim D_{\mathbb{Z},\sigma,c}$.

1: $z^+ \leftarrow$ BaseSampler()
2: $\mathbf{b} \leftarrow \mathbf{U}(\{0, 1\})$
3: $\mathbf{z} \leftarrow \mathbf{b} + (2\mathbf{b} - 1)\mathbf{z}^+$
4: $\mathbf{x} \leftarrow -\frac{(\mathbf{z}-\mathbf{c})^2}{2\sigma^2} + \frac{(\mathbf{z}^+)^2}{2\sigma^2_{\mathbf{max}}}$
5: return z with probability $\frac{\sigma_{min}}{\sigma} \cdot \exp(x)$, otherwise restart;

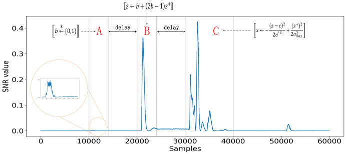
Protected SamplerZ(σ, c)

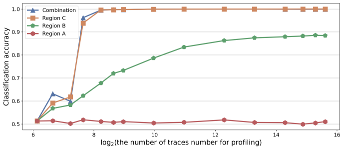
Input: c and $\sigma \in (\sigma_{min}, \sigma_{max})$.
Output: $z \sim D_{\mathbb{Z},\sigma,c}$.

1: $c' \leftarrow c - \lfloor c \rfloor$
2: $z^+ \leftarrow$ BaseSampler()
3: $(\tilde{t}[0], \ldots, \tilde{t}[15]) \leftarrow (2, 1, 1, 2, 2, 1, 1, 2, 2, 1, 2, 1, 1, 2, 1, 2)$
4: $t \leftarrow U(\{0, \ldots, 15\})$
5: $b \leftarrow \tilde{t}[t]$
6: $(\tilde{c}[0], \tilde{c}[1], \tilde{c}[2]) \leftarrow (0, c', 1 - c')$
7: $(\tilde{z}[0], \tilde{z}[1], \tilde{z}[2]) \leftarrow (0, \lfloor c \rfloor - z^+, \lfloor c \rfloor + 1 + z^+)$
8: $x \leftarrow -\frac{(z^+ + \tilde{c}[b])^2}{2\sigma^2} + \frac{(z^+)^2}{2\sigma^2_{max}}$
9: return $\tilde{z}[b]$ with probability $\frac{\sigma_{min}}{\sigma} \cdot \exp(x)$, otherwise restart;

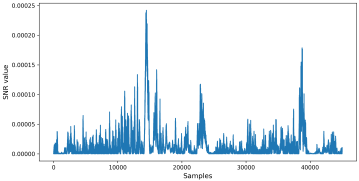
Effectiveness

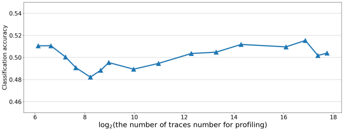
Unprotected integer sampler





Protected integer sampler





Conclusion

Conclusion

We provide an improved power analysis for Falcon.

- a new effective key recovery using the half Gaussian leakage within the base sampler
- the first side-channel analysis on Falcon taking the sign leakage into account.
- the above attacks also working with imperfect classification
- our attacks also working for the Mitaka signature scheme.

Conclusion

We provide an improved power analysis for Falcon.

- a new effective key recovery using the half Gaussian leakage within the base sampler
- the first side-channel analysis on Falcon taking the sign leakage into account.
- the above attacks also working with imperfect classification
- our attacks also working for the Mitaka signature scheme.

With the post-quantum standardization and migration underway, the side-channel security of post-quantum schemes needs more investigations.

Thank you!