Thorough Power Analysis on Falcon Gaussian Samplers and Practical Countermeasure

PKC 2025

Speaker: Haoxiang Jin

Xiuhan Lin, Shiduo Zhang, Yang Yu, Weijia Wang,
Qidi You, Ximing Xu, Xiaoyun Wang

Overview

This work mainly focuses on the side-channel security of Falcon

- further refines the key recovery of [ZLYW23][1]: $\downarrow 85\%$
- gives complete power analysis for half Gaussian leakage and sign leakage existing in Falcon's integer Gaussian sampler
- proposes effective and easy-to-implement countermeasures against both leakages

[1] [ZLYW23]: Improved Power Analysis Attacks on Falcon. Zhang, Lin, Yu and Wang.

Outline

- Background
- Further improvements of [ZLYW23]
- Complete analysis of half Gaussian and sign leakages
- Countermeasures against two leakages

Background

Falcon

Falcon[2] is one of the three post-quantum signature schemes selected by NIST for standardization.

[2] https://csrc.nist.gov/projects/post-quantum-cryptography/selected-algorithms

Falcon

Falcon[2] is one of the three post-quantum signature schemes selected by NIST for standardization.

Falcon has competitive overall performance especially the smallest communication cost (sizes of public key + signature) among other three selected signatures.

[2] https://csrc.nist.gov/projects/post-quantum-cryptography/selected-algorithms

Falcon

Falcon[2] is one of the three post-quantum signature schemes selected by NIST for standardization.

Falcon has competitive overall performance especially the smallest communication cost (sizes of public key + signature) among other three selected signatures.

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

[2] https://csrc.nist.gov/projects/post-quantum-cryptography/selected-algorithms

Hash-and-sign construction

Evolution: GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

[3][NR06]: Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Nguyen and Regev.

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

Hash-and-sign construction

Evolution: GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

Early constructions (GGH, NTRUSign)

- signing: use deterministic algorithm to find close vector
- the distribution of signatures leaks information of $\mathbf{B}$, Insecure![3]

[3] [NR06]: Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Nguyen and Regev.

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

Hash-and-sign construction

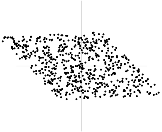
Evolution: GGH, NTRUSign $\to$ GPV $\to$ Falcon

Early constructions (GGH, NTRUSign)

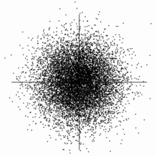
- signing: use deterministic algorithm to find close vector
- the distribution of signatures leaks information of $\mathbf{B}$, Insecure![3]

[GPV08][4] proposed a provably secure hash-and-sign framework.

- signing $\Leftrightarrow$ lattice Gaussian sampling (trapdoor sampler)



$\Rightarrow$



[3] [NR06]: Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Nguyen and Regev.

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

Hash-and-sign construction

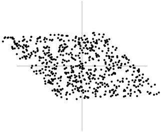
Evolution: GGH, NTRUSign $\rightarrow$ GPV $\rightarrow$ Falcon

Early constructions (GGH, NTRUSign)

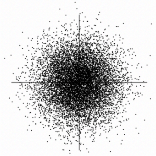
- signing: use deterministic algorithm to find close vector
- the distribution of signatures leaks information of $\mathbf{B}$, Insecure![3]

[GPV08][4] proposed a provably secure hash-and-sign framework.

- signing $\Leftrightarrow$ lattice Gaussian sampling (trapdoor sampler)



$\Rightarrow$



Falcon $=$ GPV $+$ NTRU lattices $+$ Fast Fourier Gaussian sampler (FFO)

[3] [NR06]: Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Nguyen and Regev.

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

Falcon's integer Gaussian samplers

$$\begin{array}{c} \text{FFOSampler:} \\ \mathbf{s} \leftarrow D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}} \end{array} \longleftarrow \begin{array}{c} \text{SamplerZ:} \\ z \leftarrow D_{\mathbb{Z},\sigma',c} \end{array} \longleftarrow \begin{array}{c} \text{BaseSampler:} \\ z^{+} \leftarrow D^{+}_{\mathbb{Z},\sigma_{\max},0} \end{array}$$

Klein-GPV sampler

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

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

$\mathbf{v}_n \leftarrow \mathbf{0}, \mathbf{c}_n \leftarrow \mathbf{c}$

for $i = n-1, \cdots, 0$ **do**

  $d_i = \langle \mathbf{c}_i, \tilde{\mathbf{b}}_i \rangle / \|\tilde{\mathbf{b}}_i\|^2$

  $z_i \leftarrow D_{\mathbb{Z},\sigma_i,d_i}$ where $\sigma_i = \sigma / \|\tilde{\mathbf{b}}_i\|$

  $\mathbf{c}_{i-1} \leftarrow \mathbf{c}_i - z_i \mathbf{b}_i, \mathbf{v}_{i-1} \leftarrow \mathbf{v}_i + z_i \mathbf{b}_i$

return $\mathbf{v}_0$

Falcon's integer Gaussian samplers

$$\begin{array}{|c|}\hline \text{FFOSampler:} \\ \mathbf{s} \leftarrow D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}} \\ \hline\end{array} \longleftarrow \begin{array}{|c|}\hline \text{SamplerZ:} \\ z \leftarrow D_{\mathbb{Z},\sigma',c} \\ \hline\end{array} \longleftarrow \begin{array}{|c|}\hline \text{BaseSampler:} \\ z^+ \leftarrow D^+_{\mathbb{Z},\sigma_{\max},0} \\ \hline\end{array}$$

SamplerZ

Input: A center c and $\sigma \in [\sigma_{\min}, \sigma_{\max}]$

Output: An integer z derived from a distribution close to $D_{\mathbb{Z},\sigma,c}$

$r \leftarrow c - \lfloor c \rfloor$, $ccs \leftarrow \sigma_{\min}/\sigma'$

$z^+ \leftarrow \text{BaseSampler}()$

$b \xleftarrow{\$} \{0, 1\}$

$z \leftarrow b + (2b - 1)z^+$

$x \leftarrow \frac{(z-r)^2}{2\sigma^2} - \frac{(z^+)^2}{2\sigma_{\max}^2}$

return $z + \lfloor c \rfloor$ if $\text{BerExp}(x, ccs) = 1$, otherwise restart.

Falcon's integer Gaussian samplers

$$\begin{tikzcd} \boxed{\begin{array}{c}\text{FFOSampler:}\\ \mathbf{s} \leftarrow D_{\mathcal{L}(\mathbf{B}),\sigma,\mathbf{c}}\end{array}} & \boxed{\begin{array}{c}\text{SamplerZ:}\\ z \leftarrow D_{\mathbb{Z},\sigma',c}\end{array}} \arrow[l] & \boxed{\begin{array}{c}\textcolor{red}{\text{BaseSampler:}}\\ z^+ \leftarrow D^+_{\mathbb{Z},\sigma_{\max},0}\end{array}} \arrow[l] \end{tikzcd}$$

BaseSampler

Input: -

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

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

$u \xleftarrow{\$} \{0,1\}^{72}$

$z^+ \leftarrow 0$

for $i = 0, \cdots, 17$ **do**

  $z^+ \leftarrow z^+ + [\![u < \mathsf{RCDT}[i]]\!]$

return z^+

Half Gaussian leakage

BaseSampler

Input: -

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

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

Half Gaussian leakage [GMRR22][5]

One can classify $z^+ = 0$ or $z^+ \neq 0$ by simple power analysis against the comparison of $[\![u < \mathsf{RCDT}[i]]\!]$.

[5] [GMRR22]: The Hidden Parallelepiped is Back Again: Power Analysis Attacks on Falcon. Guerreau, Martinelli, Ricosset and Rossi.

Sign leakage

SamplerZ

Input: A center c and $\sigma \in [\sigma_{\min}, \sigma_{\max}]$

Output: An integer z derived from a distribution close to $D_{\mathbb{Z},\sigma,c}$

1: $r \leftarrow c - \lfloor c \rfloor$, $ccs \leftarrow \sigma_{\min}/\sigma'$
2: $z^+ \leftarrow$ BaseSampler()
3: $b \xleftarrow{\$} \{0,1\}$
4: $z \leftarrow b + (2b-1)z^+$
5: $x \leftarrow \frac{(z-r)^2}{2\sigma^2} - \frac{(z^+)^2}{2\sigma_{\max}^2}$
6: **return** $z + \lfloor c \rfloor$ if BerExp$(x, ccs) = 1$, otherwise restart.

Sign leakage [ZLYW23][6]

One can classify b by template attacks against the operations $[\![b \xleftarrow{\$} \{0,1\}]\!]$, $[\![z \leftarrow b + (2b-1)z^+]\!]$ and $[\![x \leftarrow \frac{(z-r)^2}{2\sigma^2} - \frac{(z^+)^2}{2\sigma_{\max}^2}]\!]$.

[6] [ZLYW23]: Improved Power Analysis Attacks on Falcon. Zhang, Lin, Yu and Wang.

Further improvements of [ZLYW23]

Refining the learning with NTRU symplecticity

Due to NTRU symplecticity [GHN06][7], four rows of Falcon key satisfy:

$$\frac{\mathbf{b}_0^*}{\|\mathbf{b}_0^*\|} = \frac{\mathbf{b}_{n/2}^*}{\|\mathbf{b}_{n/2}^*\|} \cdot \mathbf{P} = -\frac{\mathbf{b}_{3n/2-1}^*}{\|\mathbf{b}_{3n/2-1}^*\|} \cdot \mathbf{P} \cdot \mathbf{J} \cdot \mathbf{Q} = \frac{\mathbf{b}_{2n-1}^*}{\|\mathbf{b}_{2n-1}^*\|} \cdot \mathbf{J} \cdot \mathbf{Q}$$

- $\mathbf{P} = \begin{pmatrix} & -\mathbf{I}_{n/2} & & \\ \mathbf{I}_{n/2} & & & \\ & & & -\mathbf{I}_{n/2} \\ & & \mathbf{I}_{n/2} & \end{pmatrix}$
- $\mathbf{J}$ is a $2n \times 2n$ reversed identity matrix, $\mathbf{Q} = \begin{pmatrix} -\mathbf{I}_n & \\ & \mathbf{I}_n \end{pmatrix}$

[7] [GHN06]: Symplectic Lattice Reduction and NTRU. Gama, Howgrave-Graham and Nguyen.

Due to NTRU symplecticity [GHN06][7], four rows of Falcon key satisfy:

$$\frac{\mathbf{b}_0^*}{\|\mathbf{b}_0^*\|} = \frac{\mathbf{b}_{n/2}^*}{\|\mathbf{b}_{n/2}^*\|} \cdot \mathbf{P} = -\frac{\mathbf{b}_{3n/2-1}^*}{\|\mathbf{b}_{3n/2-1}^*\|} \cdot \mathbf{P} \cdot \mathbf{J} \cdot \mathbf{Q} = \frac{\mathbf{b}_{2n-1}^*}{\|\mathbf{b}_{2n-1}^*\|} \cdot \mathbf{J} \cdot \mathbf{Q}$$

- $\mathbf{P} = \begin{pmatrix} & -\mathbf{I}_{n/2} & & \\ \mathbf{I}_{n/2} & & & \\ & & & -\mathbf{I}_{n/2} \\ & & \mathbf{I}_{n/2} & \end{pmatrix}$
- $\mathbf{J}$ is a $2n \times 2n$ reversed identity matrix, $\mathbf{Q} = \begin{pmatrix} -\mathbf{I}_n & \\ & \mathbf{I}_n \end{pmatrix}$

One trace contributes more information ($4\times$) compared with [ZLYW23].

[7] [GHN06]: Symplectic Lattice Reduction and NTRU. Gama, Howgrave-Graham and Nguyen.

Combining with lattice decoding technique

We correct errors from the approximation by using probability-based Prest's decoding technique [Pre23] [8] [LSZ+24] [9].

	Half Gaussian leakage	Sign leakage	Both leakages
[ZLYW23]	220,000	170,000	45,000
This work	27,500	25,000	6,500
Vs.	$\downarrow 88\%$	$\downarrow 85\%$	$\downarrow 86\%$

[8] [Pre23]: A Key-Recovery Attack against Mitaka in the t-Probing Model. Thomas Prest.

[9] [LSZ+24]: Cryptanalysis of the Peregrine Lattice-Based Signature Scheme. Lin, Suzuki, Zhang et al.

Complete analysis of half Gaussian and sign leakages

Complete power analysis against two leakages

We identify new sources of two existing power leakages and then give complete analysis against them.

- target: SamplerZ (Falcon reference implementation)
- exploit: half Gaussian leakage [GMRR22] and sign leakage [ZLYW23]
- approach: template attack
- platform: Chipwhisperer-Lite

Complete power analysis for half Gaussian leakage

For half Gaussian leakage:

- original sources: [GMRR22]
- new sources: this work

SamplerZ

Input: A center c and $\sigma \in [\sigma_{\min}, \sigma_{\max}]$

Output: An integer z derived from a distribution close to $D_{\mathbb{Z},\sigma,c}$

$r \leftarrow c - \lfloor c \rfloor$, $ccs \leftarrow \sigma_{\min}/\sigma'$

$z^+ \leftarrow \text{BaseSampler}()$

$b \xleftarrow{\$} \{0,1\}$

$z \leftarrow b + (2b-1)z^+$

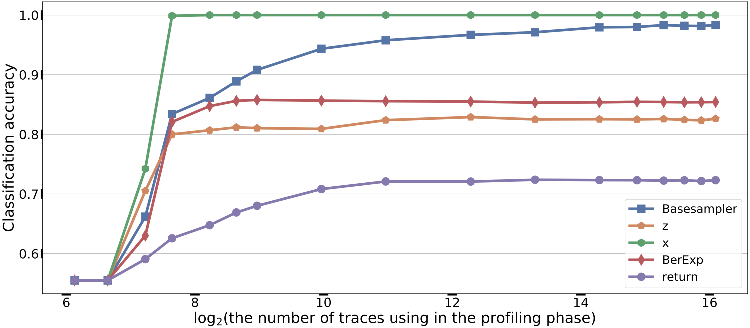
$x \leftarrow \frac{(z-r)^2}{2\sigma^2} - \frac{(z^+)^2}{2\sigma_{\max}^2}$

return $z + \lfloor c \rfloor$ if $\text{BerExp}(x, ccs) = 1$, otherwise restart.

Complete analysis = original sources + new sources

Security evaluations

For half Gaussian leakage, the classification accuracy of single trace attacks is:



Complete power analysis for sign leakage

For sign leakage:

- original sources: [ZLYW23]
- new sources: this work

SamplerZ

Input: A center c and $\sigma \in [\sigma_{\min}, \sigma_{\max}]$

Output: An integer z derived from a distribution close to $D_{\mathbb{Z},\sigma,c}$

$r \leftarrow c - \lfloor c \rfloor$, $ccs \leftarrow \sigma_{\min}/\sigma'$

$z^+ \leftarrow \text{BaseSampler}()$

$b \xleftarrow{\$} \{0, 1\}$

$z \leftarrow b + (2b - 1)z^+$

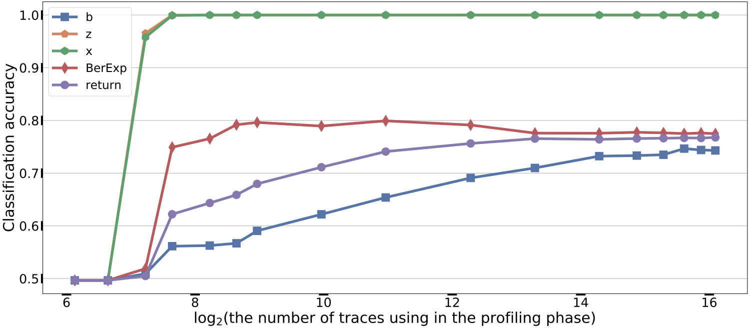
$x \leftarrow \frac{(z-r)^2}{2\sigma^2} - \frac{(z^+)^2}{2\sigma_{\max}^2}$

return $z + \lfloor c \rfloor$ if $\text{BerExp}(x, ccs) = 1$, otherwise restart.

Complete analysis = original sources + new sources

Security evaluations

For sign leakage, the classification accuracy of single trace attacks is:



Countermeasures against two leakages

Countermeasures against half Gaussian leakage

Validation for the countermeasures of [GMRR22]:

- tricks: $\{0, 255\} \Rightarrow \{0, 1\}$
- platform: Chipwhisperer-Lite
- the classification accuracy is still at least 97%

Countermeasures against half Gaussian leakage

Validation for the countermeasures of [GMRR22]:

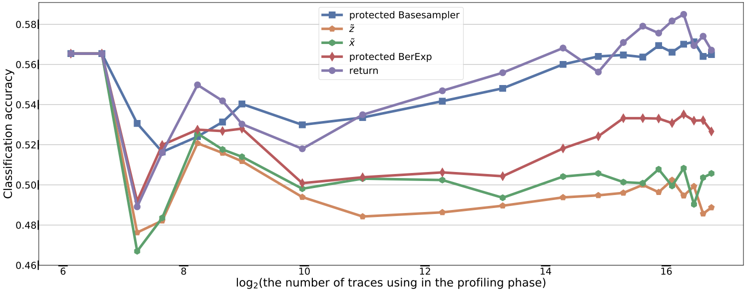
- tricks: $\{0, 255\} \Rightarrow \{0, 1\}$
- platform: Chipwhisperer-Lite
- the classification accuracy is still at least 97%

Our countermeasures

1. $\{0, 255\} \Rightarrow \{0, 1\} \Rightarrow \{1, 2\}$
2. multiple sampling
3. the traversal of $z^+ \in \{0, \cdots, 18\}$
4. table look-ups with index z^+

Security evaluations

For half Gaussian leakage, the classification accuracy is at most $\approx 58\%$.



When the accuracy is $\leq 65\%$, the required traces for full key recovery are much more than 10 million. Impractical![10]

[10] see Figure 5 of [ZLYW23].

Countermeasures against sign leakage

Validation for the countermeasures of [ZLYW23]:

- tricks: $\{0, 1\} \Rightarrow \{1, 2\}$
- platform: Chipwhisperer-Lite
- the classification accuracy for the computation of x is still 75%

Countermeasures against sign leakage

Validation for the countermeasures of [ZLYW23]:

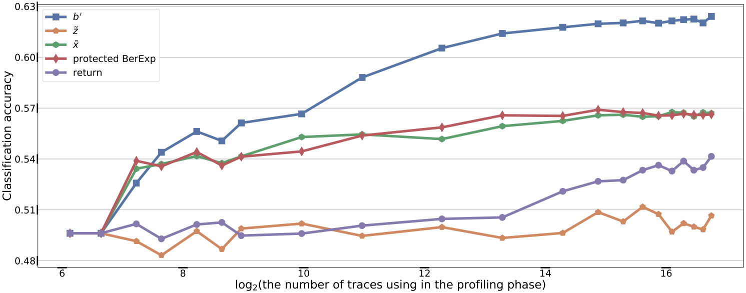
- tricks: $\{0, 1\} \Rightarrow \{1, 2\}$
- platform: Chipwhisperer-Lite
- the classification accuracy for the computation of x is still 75%

Our countermeasures

1. $\{0, 1\} \Rightarrow \{1, 2\}$
2. the traversal of $b' \in \{1, 2\}$
3. table look-ups with index b'

Security evaluations

For sign leakage, the classification accuracy is at most $\approx 62\%$.



When the accuracy is $\leq 65\%$, the required traces for full key recovery are much more than 10 million. Impractical![11]

[11] see Figure 12 of [ZLYW23].

Performance evaluations

We also report benchmarks for Falcon's signing (SD: dynamic mode, ST: tree mode) with countermeasures

- based on the reference implementation of Falcon
- platform: Intel Core i5-1135G7 CPU
- compilation: `Clang-10.0.0` with cflags `-O0`

Claimed Security	Falcon-512		Falcon-1024	
	SD	ST	SD	ST
Unprotected (ms)	6.7	3.1	14.8	6.5
Protected (ms)	24.5	20.5	49.4	41.0
Vs.	3.7×	6.6×	3.3×	6.3×
Unprotected (Mcycles)	16.6	7.3	35.6	15.7
Protected (Mcycles)	58.7	49.9	119.6	99.4
Vs.	3.5×	6.8×	3.4×	6.3×

Conclusion

Conclusion

This work gives complete power analysis for Falcon's integer Gaussian sampler from the perspective of attacks and protections.

Our source code is available at

https://github.com/lxhcrypto/FalconAnalysis

With the deployment underway, the side-channel security of post-quantum schemes requires more investigations.

Thank you!