On the Side-Channel Resistance of UOV

Survey of Physical Attacks and Recent Developments

Thomas Aulbach[1]

NIST PQC Seminars , 07.07.2023

[1]Universität Regensburg, Regensburg, Germany

Outline

1. UOV from Two Perspectives
2. Fault Attacks
   - Skip Random Sampling of Vinegar Variables [SK20]
   - Bit-Flip in Central Map [FKN+22]
3. Side Channel Attacks
   - Horizontal SCA on Linear Transformation [PSK+18]
   - Template Attack on Evaluation of Vinegar Variables [ACK+23]
4. Takeaways

UOV from Two Perspectives

UOV - a Remarkable Candidate

UOV stands out, since

- it is a comparably old scheme with 25 years of cryptanalysis
- many current (and past) multivariate signature schemes are modifications of it

[1] https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/G0DoD7lkGPk

UOV - a Remarkable Candidate

UOV stands out, since

- it is a comparably old scheme with 25 years of cryptanalysis
- many current (and past) multivariate signature schemes are modifications of it

NIST would like submissions for signature schemes that:[1]

[1] https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/G0DoD7lkGPk

UOV - a Remarkable Candidate

UOV stands out, since

- it is a comparably old scheme with 25 years of cryptanalysis
- many current (and past) multivariate signature schemes are modifications of it

NIST would like submissions for signature schemes that:[1]

- 'are **not based on structured lattices**' ✓

[1] https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/G0DoD7lkGPk

UOV - a Remarkable Candidate

UOV stands out, since

- it is a comparably old scheme with 25 years of cryptanalysis
- many current (and past) multivariate signature schemes are modifications of it

NIST would like submissions for signature schemes that:[1]

- 'are **not based on structured lattices**' ✓
- 'have **short signatures**' ✓

[1] https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/G0DoD7lkGPk

UOV - a Remarkable Candidate

UOV stands out, since

- it is a comparably old scheme with 25 years of cryptanalysis
- many current (and past) multivariate signature schemes are modifications of it

NIST would like submissions for signature schemes that:[1]

- 'are **not based on structured lattices**' ✓
- 'have **short signatures**' ✓
- 'and **fast verification**' ✓

[1] https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/G0DoD7lkGPk

UOV - a Remarkable Candidate

UOV stands out, since

- it is a comparably old scheme with 25 years of cryptanalysis
- many current (and past) multivariate signature schemes are modifications of it

NIST would like submissions for signature schemes that:[1]

- 'are **not based on structured lattices**' ✓
- 'have **short signatures**' ✓
- 'and **fast verification**' ✓
- 'e.g., **UOV**' ✓

[1] https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/G0DoD7lkGPk

Comparison with Dilithium

Oil and Vinegar: Modern Parameters and Implementations[2]

Key sizes and performance data

Signature Scheme	public key	secret key (Bytes)	signature	KeyGen	Sign (Cycles)	Verify
ov-Ip	278 432	237 912	128	2 903 434	105 324	90 336
ov-Ip-pk**c**	43 576	237 912	128	2 858 724	105 324	224 006
ov-Ip-pk**c**-sk**c**	43 576	64	128	2 848 774	1 876 442	224 006
Dilithium2	1 312	2 544	2 420	124 031	333 013	118 412

[2] Beullens, W., Chen, M. S., Hung, S. H., Kannwischer, M. J., Peng, B. Y., Shih, C. J., and Yang, B. Y. (2023). Oil and Vinegar: Modern Parameters and Implementations. IACR TCHES, 321-365.

Multivariate Cryptography

Signatures from multivariate quadratic equations:

- Key objects are multivariate quadratic maps $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$

Signatures from multivariate quadratic equations:

- Key objects are multivariate quadratic maps $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$
- $\mathcal{P}$ consists of m homogeneous quadratic polynomials

$$p_k(\boldsymbol{x}) = \sum_{1 \le i \le j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \boldsymbol{x} = (x_1, \ldots, x_n)^\top \in \mathbb{F}_q^n$$

Signatures from multivariate quadratic equations:

- Key objects are multivariate quadratic maps $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$
- $\mathcal{P}$ consists of m homogeneous quadratic polynomials

$$p_k(\mathbf{x}) = \sum_{1 \leq i \leq j \leq n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \mathbf{x} = (x_1, \ldots, x_n)^\top \in \mathbb{F}_q^n$$

- **Signing** d in a nutshell: For $\mathbf{t} = H(d) \in \mathbb{F}_q^m$, find $\mathbf{s} \in \mathbb{F}_q^n$, such that $\mathcal{P}(\mathbf{s}) = \mathbf{t}$

Signatures from multivariate quadratic equations:

- Key objects are multivariate quadratic maps $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$
- $\mathcal{P}$ consists of m homogeneous quadratic polynomials

$$p_k(\mathbf{x}) = \sum_{1 \le i \le j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \mathbf{x} = (x_1, \dots, x_n)^\top \in \mathbb{F}_q^n$$

- **Signing** d in a nutshell: For $\boldsymbol{t} = H(d) \in \mathbb{F}_q^m$, find $\boldsymbol{s} \in \mathbb{F}_q^n$, such that $\mathcal{P}(\boldsymbol{s}) = \boldsymbol{t}$
  - In general this is really difficult

Multivariate Cryptography

Signatures from multivariate quadratic equations:

- Key objects are multivariate quadratic maps $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$
- $\mathcal{P}$ consists of m homogeneous quadratic polynomials

$$p_k(\mathbf{x}) = \sum_{1 \leq i \leq j \leq n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \mathbf{x} = (x_1, \ldots, x_n)^\top \in \mathbb{F}_q^n$$

- **Signing** d in a nutshell: For $\boldsymbol{t} = H(d) \in \mathbb{F}_q^m$, find $\boldsymbol{s} \in \mathbb{F}_q^n$, such that $\mathcal{P}(\boldsymbol{s}) = \boldsymbol{t}$
  - In general this is really difficult
  - Include a trapdoor that can only be used with the secret key

Signatures from multivariate quadratic equations:

- Key objects are multivariate quadratic maps $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$
- $\mathcal{P}$ consists of m homogeneous quadratic polynomials

$$p_k(\mathbf{x}) = \sum_{1 \le i \le j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \mathbf{x} = (x_1, \ldots, x_n)^\top \in \mathbb{F}_q^n$$

- **Signing** d in a nutshell: For $\boldsymbol{t} = H(d) \in \mathbb{F}_q^m$, find $\boldsymbol{s} \in \mathbb{F}_q^n$, such that $\mathcal{P}(\boldsymbol{s}) = \boldsymbol{t}$
  - In general this is really difficult
  - Include a trapdoor that can only be used with the secret key
- **Verify** if $\mathcal{P}(\boldsymbol{s}) = \boldsymbol{t}$ really holds

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- $\mathcal{P} = \mathcal{F} \circ T$, where $\mathcal{F}$ is structured and easy to invert and T is a linear transformation

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- $\mathcal{P} = \mathcal{F} \circ T$, where $\mathcal{F}$ is structured and easy to invert and T is a linear transformation
- $\mathcal{F}$ consists of m homogeneous quadratic polynomials

$$f_k(\mathbf{x}) = \sum_{1 \le i \le j \le v} \alpha_{i,j}^{(k)} x_i x_j + \sum_{1 \le i \le v < j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$$

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- $\mathcal{P} = \mathcal{F} \circ T$, where $\mathcal{F}$ is structured and easy to invert and T is a linear transformation
- $\mathcal{F}$ consists of m homogeneous quadratic polynomials

$$f_k(\mathbf{x}) = \sum_{1 \le i \le j \le v} \alpha_{i,j}^{(k)} x_i x_j + \sum_{1 \le i \le v < j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$$

- Sort coefficients to matrices $F^{(k)}$ such that $f_k(\mathbf{x}) = \mathbf{x}^\top F^{(k)} \mathbf{x}$

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- $\mathcal{P} = \mathcal{F} \circ T$, where $\mathcal{F}$ is structured and easy to invert and T is a linear transformation
- $\mathcal{F}$ consists of m homogeneous quadratic polynomials

$$f_k(\mathbf{x}) = \sum_{1 \le i \le j \le v} \alpha_{i,j}^{(k)} x_i x_j + \sum_{1 \le i \le v < j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$$

- Sort coefficients to matrices $F^{(k)}$ such that $f_k(\mathbf{x}) = \mathbf{x}^\top F^{(k)} \mathbf{x}$

$$\begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}^\top \begin{pmatrix} \alpha_{1,1}^{(k)} & \cdots & \alpha_{1,v}^{(k)} & \alpha_{1,v+1}^{(k)} & \cdots & \alpha_{1,n}^{(k)} \\ 0 & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \alpha_{v,v}^{(k)} & \alpha_{v,v+1}^{(k)} & \cdots & \alpha_{v,n}^{(k)} \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}$$

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- $\mathcal{P} = \mathcal{F} \circ T$, where $\mathcal{F}$ is structured and easy to invert and T is a linear transformation
- $\mathcal{F}$ consists of m homogeneous quadratic polynomials

$$f_k(\mathbf{x}) = \sum_{1 \le i \le j \le v} \alpha_{i,j}^{(k)} x_i x_j + \sum_{1 \le i \le v < j \le n} \alpha_{i,j}^{(k)} x_i x_j, \text{ where } \mathbf{x} = (x_1, \dots, x_n) \in \mathbb{F}_q^n$$

- Sort coefficients to matrices $F^{(k)}$ such that $f_k(\mathbf{x}) = \mathbf{x}^\top F^{(k)} \mathbf{x}$

$$\begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}^\top \begin{pmatrix} \alpha_{1,1}^{(k)} & \dots & \alpha_{1,v}^{(k)} & \alpha_{1,v+1}^{(k)} & \dots & \alpha_{1,n}^{(k)} \\ 0 & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \alpha_{v,v}^{(k)} & \alpha_{v,v+1}^{(k)} & \dots & \alpha_{v,n}^{(k)} \\ 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix} = l_1^{(k)} \cdot y_1 + \dots + l_m^{(k)} \cdot y_m + c^{(k)}$$

- Fix and insert vinegar variables $\tilde{v}_i$ to get m linear equations in m oil variables

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- Compute between pk = $\mathcal{P}$ and sk = $(\mathcal{F}, T)$ with

$$P^{(k)} = T^\top F^{(k)} T$$

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- Compute between pk = $\mathcal{P}$ and sk = $(\mathcal{F}, T)$ with

$$P^{(k)} = T^\top F^{(k)} T$$

- Visualization of signing $\mathbf{t}$

$$\mathbf{t} = \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} v_1 \\ \vdots \\ v_{n-m} \\ y_1 \\ \vdots \\ y_m \end{pmatrix} = \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

Two Descriptions of UOV in the Literature (1/2)

UOV with hidden central map $\mathcal{F}$

- Compute between pk = $\mathcal{P}$ and sk = $(\mathcal{F}, T)$ with

$$P^{(k)} = T^\top F^{(k)} T$$

- Visualization of signing **t**

$$\mathbf{t} = \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix} \xrightarrow{\quad \mathcal{F}^{-1} \quad} \begin{pmatrix} v_1 \\ \vdots \\ v_{n-m} \\ y_1 \\ \vdots \\ y_m \end{pmatrix} = \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{\quad T^{-1} \quad} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

- T has block matrix structure $T = \begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix}$

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\boldsymbol{o}) = 0 \text{ for all } \boldsymbol{o} \in O$$

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\mathbf{o}) = 0 \text{ for all } \mathbf{o} \in O$$

- The map $\mathcal{P}'(\mathbf{x}, \mathbf{y}) := \mathcal{P}(\mathbf{x} + \mathbf{y}) - \mathcal{P}(\mathbf{x}) - \mathcal{P}(\mathbf{y})$ is bilinear and symmetric

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\boldsymbol{o}) = 0 \text{ for all } \boldsymbol{o} \in O$$

- The map $\mathcal{P}'(\boldsymbol{x}, \boldsymbol{y}) := \mathcal{P}(\boldsymbol{x} + \boldsymbol{y}) - \mathcal{P}(\boldsymbol{x}) - \mathcal{P}(\boldsymbol{y})$ is bilinear and symmetric

Signing strategy:

- Generate random $\boldsymbol{v} \in \mathbb{F}_q^n$

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\boldsymbol{o}) = 0 \text{ for all } \boldsymbol{o} \in O$$

- The map $\mathcal{P}'(\boldsymbol{x}, \boldsymbol{y}) := \mathcal{P}(\boldsymbol{x} + \boldsymbol{y}) - \mathcal{P}(\boldsymbol{x}) - \mathcal{P}(\boldsymbol{y})$ is bilinear and symmetric

Signing strategy:

- Generate random $\boldsymbol{v} \in \mathbb{F}_q^n$
- Solve $\mathcal{P}(\boldsymbol{v} + \boldsymbol{o}) = \mathcal{P}(\boldsymbol{v}) + \mathcal{P}(\boldsymbol{o}) + \mathcal{P}'(\boldsymbol{v}, \boldsymbol{o}) = \mathbf{t}$ for $\boldsymbol{o} \in O$.

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\boldsymbol{o}) = 0 \text{ for all } \boldsymbol{o} \in O$$

- The map $\mathcal{P}'(\boldsymbol{x}, \boldsymbol{y}) := \mathcal{P}(\boldsymbol{x} + \boldsymbol{y}) - \mathcal{P}(\boldsymbol{x}) - \mathcal{P}(\boldsymbol{y})$ is bilinear and symmetric

Signing strategy:

- Generate random $\boldsymbol{v} \in \mathbb{F}_q^n$
- Solve $\mathcal{P}(\boldsymbol{v} + \boldsymbol{o}) = \mathcal{P}(\boldsymbol{v}) + \mathcal{P}(\boldsymbol{o}) + \mathcal{P}'(\boldsymbol{v}, \boldsymbol{o}) = \mathbf{t}$ for $\boldsymbol{o} \in O$.
  $\rightarrow$ Computing $\mathcal{P}(\boldsymbol{v})$ implies the insertion of the vinegar variables into the quadratic map

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\boldsymbol{o}) = 0 \text{ for all } \boldsymbol{o} \in O$$

- The map $\mathcal{P}'(\boldsymbol{x}, \boldsymbol{y}) := \mathcal{P}(\boldsymbol{x} + \boldsymbol{y}) - \mathcal{P}(\boldsymbol{x}) - \mathcal{P}(\boldsymbol{y})$ is bilinear and symmetric

Signing strategy:

- Generate random $\boldsymbol{v} \in \mathbb{F}_q^n$
- Solve $\mathcal{P}(\boldsymbol{v} + \boldsymbol{o}) = \mathcal{P}(\boldsymbol{v}) + \mathcal{P}(\boldsymbol{o}) + \mathcal{P}'(\boldsymbol{v}, \boldsymbol{o}) = \mathbf{t}$ for $\boldsymbol{o} \in O$.
  $\rightarrow$ Computing $\mathcal{P}(\boldsymbol{v})$ implies the insertion of the vinegar variables into the quadratic map
  $\rightarrow$ Solving $\mathcal{P}'(\boldsymbol{v}, \boldsymbol{o}) = \mathbf{t} - \mathcal{P}(\boldsymbol{v})$ means solving a system with m variables in m equations

Two Descriptions of UOV in the Literature (2/2)

UOV with secret oil space

- Define $\mathcal{P}$ such that it vanishes on secret linear oil space $O \subset \mathbb{F}_q^n$ of dimension m, i.e.

$$\mathcal{P}(\boldsymbol{o}) = 0 \text{ for all } \boldsymbol{o} \in O$$

- The map $\mathcal{P}'(\boldsymbol{x}, \boldsymbol{y}) := \mathcal{P}(\boldsymbol{x} + \boldsymbol{y}) - \mathcal{P}(\boldsymbol{x}) - \mathcal{P}(\boldsymbol{y})$ is bilinear and symmetric

Signing strategy:

- Generate random $\boldsymbol{v} \in \mathbb{F}_q^n$
- Solve $\mathcal{P}(\boldsymbol{v} + \boldsymbol{o}) = \mathcal{P}(\boldsymbol{v}) + \mathcal{P}(\boldsymbol{o}) + \mathcal{P}'(\boldsymbol{v}, \boldsymbol{o}) = \mathbf{t}$ for $\boldsymbol{o} \in O$.
  $\rightarrow$ Computing $\mathcal{P}(\boldsymbol{v})$ implies the insertion of the vinegar variables into the quadratic map
  $\rightarrow$ Solving $\mathcal{P}'(\boldsymbol{v}, \boldsymbol{o}) = \mathbf{t} - \mathcal{P}(\boldsymbol{v})$ means solving a system with m variables in m equations
- The vector $\boldsymbol{s} = \boldsymbol{v} + \boldsymbol{o}$ forms a valid signature

Fault Attacks

Skip Random Sampling of Vinegar Variables

Main idea

- Skip the random sampling of vinegar values (already discussed in [HTS11][3] and [KL19][4])

$$\mathbf{t} \xrightarrow{\quad \mathcal{F}^{-1} \quad} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{\quad T^{-1} \quad} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

[3] Hashimoto, Y., Takagi, T., and Sakurai, K.: General Fault Attacks on Multivariate Public Key Cryptosystems. PQCrypto 2011

[4] Krämer, J., and Loiero, M.: Fault Attacks on UOV and Rainbow. COSADE 2019

Skip Random Sampling of Vinegar Variables

Main idea

- Skip the random sampling of vinegar values (already discussed in [HTS11][3] and [KL19][4])

$$\mathbf{t} \xrightarrow{\quad\mathcal{F}^{-1}\quad} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{\quad T^{-1}\quad} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

- Solution to $\mathcal{F}^{-1}$ are the randomly generated vinegar values $\mathbf{v} = \left(v_1, \ldots, v_{n-m}\right)^\top$ and the computed oil variables $\mathbf{y} = \left(y_1, \ldots, y_m\right)^\top$

[3] Hashimoto, Y., Takagi, T., and Sakurai, K.: General Fault Attacks on Multivariate Public Key Cryptosystems. PQCrypto 2011

[4] Krämer, J., and Loiero, M.: Fault Attacks on UOV and Rainbow. COSADE 2019

Skip Random Sampling of Vinegar Variables

$$\mathbf{t} \xrightarrow{\quad \mathcal{F}^{-1} \quad} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{\quad T^{-1} \quad} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

$$\mathbf{t}^{(i)} \xrightarrow{\quad \mathcal{F}^{-1} \quad} \begin{pmatrix} \mathbf{v} \\ \mathbf{y}^{(i)} \end{pmatrix} \xrightarrow{\quad T^{-1} \quad} \begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y}^{(i)} \\ \mathbf{y}^{(i)} \end{pmatrix}$$

Skip Random Sampling of Vinegar Variables

$$\mathbf{t} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

$$\mathbf{t}^{(i)} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y}^{(i)} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y}^{(i)} \\ \mathbf{y}^{(i)} \end{pmatrix}$$

- Skip random sampling enforces reuse of $\mathbf{v}$

Skip Random Sampling of Vinegar Variables

$$\mathbf{t} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

$$\mathbf{t}^{(i)} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y}^{(i)} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y}^{(i)} \\ \mathbf{y}^{(i)} \end{pmatrix}$$

- Skip random sampling enforces reuse of $\mathbf{v}$
- We have $\begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} - \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix}$

Skip Random Sampling of Vinegar Variables

$$\mathbf{t} \xrightarrow{\quad\mathcal{F}^{-1}\quad} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{\quad T^{-1}\quad} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

$$\mathbf{t}^{(i)} \xrightarrow{\quad\mathcal{F}^{-1}\quad} \begin{pmatrix} \mathbf{v} \\ \mathbf{y}^{(i)} \end{pmatrix} \xrightarrow{\quad T^{-1}\quad} \begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y}^{(i)} \\ \mathbf{y}^{(i)} \end{pmatrix}$$

- Skip random sampling enforces reuse of $\mathbf{v}$
- We have $\begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} - \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix}$
- Repeat m times to solve for T_1 (requires m faulted signatures)

Reduce Number of Needed Faulted Signatures

In fact, one can do even better

- The vector $\begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} - \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix}$ represents an oil vector, i.e.

$$\mathcal{P}\begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathcal{F}\begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix}\begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathcal{F}\begin{pmatrix} \mathbf{0} \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathbf{0}$$

Reduce Number of Needed Faulted Signatures

In fact, one can do even better

- The vector $\begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} - \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix}$ represents an oil vector, i.e.

$$\mathcal{P} \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathcal{F} \begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix} \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathcal{F} \begin{pmatrix} \mathbf{0} \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathbf{0}$$

- This is easy to recognize in the oil space description, since

$$\mathbf{s}^{(i)} - \mathbf{s} = (\mathbf{v} + \mathbf{o}^{(i)}) - (\mathbf{v} + \mathbf{o}) = \mathbf{o}^{(i)} - \mathbf{o} \in O$$

Reduce Number of Needed Faulted Signatures

In fact, one can do even better

- The vector $\begin{pmatrix} \mathbf{s}_1^{(i)} \\ \mathbf{s}_2^{(i)} \end{pmatrix} - \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix}$ represents an oil vector, i.e.

$$\mathcal{P} \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathcal{F} \begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix} \begin{pmatrix} T_1 \cdot (\mathbf{y}^{(i)} - \mathbf{y}) \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathcal{F} \begin{pmatrix} \mathbf{0} \\ (\mathbf{y}^{(i)} - \mathbf{y}) \end{pmatrix} = \mathbf{0}$$

- This is easy to recognize in the oil space description, since

$$\mathbf{s}^{(i)} - \mathbf{s} = (\mathbf{v} + \mathbf{o}^{(i)}) - (\mathbf{v} + \mathbf{o}) = \mathbf{o}^{(i)} - \mathbf{o} \in O$$

- One oil vector enables key recovery in polynomial time $\rightarrow$ next slide

Algebraic Attack

Knowledge of an oil vector dramatically simplifies algebraic key recovery attacks

- For two oil vectors $\mathbf{o}_1, \mathbf{o}_2$ it holds

$$\mathcal{P}'(\mathbf{o}_1, \mathbf{o}_2) = \mathcal{P}(\mathbf{o}_1 + \mathbf{o}_2) - \mathcal{P}(\mathbf{o}_1) - \mathcal{P}(\mathbf{o}_2) = \mathbf{0} \in \mathbb{F}_q^m$$

Algebraic Attack

Knowledge of an oil vector dramatically simplifies algebraic key recovery attacks

- For two oil vectors $\mathbf{o}_1, \mathbf{o}_2$ it holds

$$\mathcal{P}'(\mathbf{o}_1, \mathbf{o}_2) = \mathcal{P}(\mathbf{o}_1 + \mathbf{o}_2) - \mathcal{P}(\mathbf{o}_1) - \mathcal{P}(\mathbf{o}_2) = \mathbf{0} \in \mathbb{F}_q^m$$

$\rightarrow$ If $\mathbf{o}_1$ and $\mathbf{o}_2$ are unknown, this is a quadratic system that is hard to solve

Algebraic Attack

Knowledge of an oil vector dramatically simplifies algebraic key recovery attacks

- For two oil vectors $\mathbf{o}_1, \mathbf{o}_2$ it holds

$$\mathcal{P}'(\mathbf{o}_1, \mathbf{o}_2) = \mathcal{P}(\mathbf{o}_1 + \mathbf{o}_2) - \mathcal{P}(\mathbf{o}_1) - \mathcal{P}(\mathbf{o}_2) = \mathbf{0} \in \mathbb{F}_q^m$$

$\rightarrow$ If $\mathbf{o}_1$ and $\mathbf{o}_2$ are unknown, this is a quadratic system that is hard to solve
$\rightarrow$ If $\mathbf{o}_1$ is known, this presents m linear equations for $\mathbf{o}_2$

Algebraic Attack

Knowledge of an oil vector dramatically simplifies algebraic key recovery attacks

- For two oil vectors $\mathbf{o}_1, \mathbf{o}_2$ it holds

$$\mathcal{P}'(\mathbf{o}_1, \mathbf{o}_2) = \mathcal{P}(\mathbf{o}_1 + \mathbf{o}_2) - \mathcal{P}(\mathbf{o}_1) - \mathcal{P}(\mathbf{o}_2) = \mathbf{0} \in \mathbb{F}_q^m$$

  $\rightarrow$ If $\mathbf{o}_1$ and $\mathbf{o}_2$ are unknown, this is a quadratic system that is hard to solve
  $\rightarrow$ If $\mathbf{o}_1$ is known, this presents m linear equations for $\mathbf{o}_2$
- With the given UOV parameters, this implies: If **two oil vectors** are known, the remaining oil space can be found in polynomial time

Algebraic Attack

Knowledge of an oil vector dramatically simplifies algebraic key recovery attacks

- For two oil vectors $\mathbf{o}_1, \mathbf{o}_2$ it holds

$$\mathcal{P}'(\mathbf{o}_1, \mathbf{o}_2) = \mathcal{P}(\mathbf{o}_1 + \mathbf{o}_2) - \mathcal{P}(\mathbf{o}_1) - \mathcal{P}(\mathbf{o}_2) = \mathbf{0} \in \mathbb{F}_q^m$$

  $\rightarrow$ If $\mathbf{o}_1$ and $\mathbf{o}_2$ are unknown, this is a quadratic system that is hard to solve
  $\rightarrow$ If $\mathbf{o}_1$ is known, this presents m linear equations for $\mathbf{o}_2$
- With the given UOV parameters, this implies: If **two oil vectors** are known, the remaining oil space can be found in polynomial time

In fact, even **one oil vector** is enough, when using modified Kipnis-Shamir attack [5]

[5] Thanks to Ward Beullens for pointing out how this attack is possible

Algebraic Attack

Knowledge of an oil vector dramatically simplifies algebraic key recovery attacks

- For two oil vectors $\mathbf{o}_1, \mathbf{o}_2$ it holds

$$\mathcal{P}'(\mathbf{o}_1, \mathbf{o}_2) = \mathcal{P}(\mathbf{o}_1 + \mathbf{o}_2) - \mathcal{P}(\mathbf{o}_1) - \mathcal{P}(\mathbf{o}_2) = \mathbf{0} \in \mathbb{F}_q^m$$

  $\rightarrow$ If $\mathbf{o}_1$ and $\mathbf{o}_2$ are unknown, this is a quadratic system that is hard to solve
  $\rightarrow$ If $\mathbf{o}_1$ is known, this presents m linear equations for $\mathbf{o}_2$

- With the given UOV parameters, this implies: If **two oil vectors** are known, the remaining oil space can be found in polynomial time

In fact, even **one oil vector** is enough, when using modified Kipnis-Shamir attack [5]
Details can be found in [ACK+23] [6]

[5] Thanks to Ward Beullens for pointing out how this attack is possible
[6] Aulbach, T., Campos, F., Krämer, J., Samardjiska, S., and Stöttinger, M.: Separating Oil and Vinegar with a Single Trace: Side-Channel Assisted Kipnis-Shamir Attack on UOV. IACR TCHES 2023

Summary of the Fault Attack [SK20]

Summary

- **Instruction skip** to reuse the vinegar variables
- **Number** of needed **faulted signatures** is reduced from m to now only **1**
- Distinguish between reuse and zero setting (analyzed in [SK20][7] and [KKT22][8])

[7] Shim, K. A., and Koo, N.: Algebraic Fault Analysis of UOV and Rainbow with the Leakage of Random Vinegar Values. IEEE Transactions on Information Forensics and Security 2020

[8] Kato, T., Kiyomura, Y., and Takagi, T.: Improving Fault Attacks on Rainbow with Fixing Random Vinegar Values. International Workshop on Security 2022

Summary of the Fault Attack [SK20]

Summary

- **Instruction skip** to reuse the vinegar variables
- **Number** of needed **faulted signatures** is reduced from m to now only **1**
- Distinguish between reuse and zero setting (analyzed in [SK20][7] and [KKT22][8])

Practicality

- Attack is simulated targeting Rainbow on an emulated ARM M4 architecture using QEMU in [AKK+22][9]

[7] Shim, K. A., and Koo, N.: Algebraic Fault Analysis of UOV and Rainbow with the Leakage of Random Vinegar Values. IEEE Transactions on Information Forensics and Security 2020

[8] Kato, T., Kiyomura, Y., and Takagi, T.: Improving Fault Attacks on Rainbow with Fixing Random Vinegar Values. International Workshop on Security 2022

[9] Aulbach, T., Kovats, T., Krämer, J., and Marzougui, S.: Recovering Rainbow's Secret Key with a First-Order Fault Attack. AfricaCrypt 2022

Summary of the Fault Attack [SK20]

Summary

- **Instruction skip** to reuse the vinegar variables
- **Number** of needed **faulted signatures** is reduced from m to now only **1**
- Distinguish between reuse and zero setting (is analyzed in [SK20] and [KKT22])

Practicality

- Attack is simulated targeting Rainbow on an emulated ARM M4 architecture using QEMU in [AKK+22]

Countermeasures

- 'Verify before output' is not possible, since faulted signature is valid
- Store old vinegar variables and only output signature if there are no large overlaps

Summary of the Fault Attack [SK20]

Summary

- **Instruction skip** to reuse the vinegar variables
- **Number** of needed **faulted signatures** is reduced from m to now only **1**
- Distinguish between reuse and zero setting (is analyzed in [SK20] and [KKT22])

Practicality

- Attack is simulated targeting Rainbow on an emulated ARM M4 architecture using QEMU in [AKK+22]

Countermeasures

- 'Verify before output' is not possible, since faulted signature is valid
- Store old vinegar variables and only output signature if there are no large overlaps

Future work

- Execute instruction skip on a target device
- Apply to various modifications of UOV

Bit-Flip in Central Map

Fault model

- Introduce a fault that changes one coefficient ${\alpha'}_{i,j}^{(k)}$ in the central map $\mathcal{F}$ (already discussed in [HTS11] and [KL19])
- Faulted coefficient is randomly chosen and attacker does not know its location

$$F'^{(k)} = \begin{pmatrix} \alpha_{1,1}^{(k)} & \dots & \alpha_{1,v}^{(k)} & \alpha_{1,v+1}^{(k)} & \dots & \alpha_{1,n}^{(k)} \\ 0 & \ddots & \vdots & \vdots & {\alpha'}_{i,j}^{(k)} & \vdots \\ 0 & 0 & \alpha_{v,v}^{(k)} & \alpha_{v,v+1}^{(k)} & \dots & \alpha_{v,n}^{(k)} \\ 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix}$$

Fault model

- Introduce a fault that changes one coefficient ${\alpha'}_{i,j}^{(k)}$ in the central map $\mathcal{F}$ (already discussed in [HTS11] and [KL19])
- Faulted coefficient is randomly chosen and attacker does not know its location

$$
\begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}^{\top}
\begin{pmatrix}
\alpha_{1,1}^{(k)} & \dots & \alpha_{1,v}^{(k)} & \alpha_{1,v+1}^{(k)} & \dots & \alpha_{1,n}^{(k)} \\
0 & \ddots & \vdots & \vdots & {\alpha'}_{i,j}^{(k)} & \vdots \\
0 & 0 & \alpha_{v,v}^{(k)} & \alpha_{v,v+1}^{(k)} & \dots & \alpha_{v,n}^{(k)} \\
0 & \dots & 0 & 0 & \dots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \dots & 0 & 0 & \dots & 0
\end{pmatrix}
\begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}
= l_1^{(k)} \cdot y_1 + \ldots + l_j^{(k)} \cdot y_j + \ldots + c^{(k)}
$$

Bit-Flip in Central Map

Fault model

- Introduce a fault that changes one coefficient ${\alpha'}^{(k)}_{i,j}$ in the central map $\mathcal{F}$ (already discussed in [HTS11] and [KL19])
- Faulted coefficient is randomly chosen and attacker does not know its location

$$
\begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}^{\top}
\begin{pmatrix}
\alpha^{(k)}_{1,1} & \dots & \alpha^{(k)}_{1,v} & \alpha^{(k)}_{1,v+1} & \dots & \alpha^{(k)}_{1,n} \\
0 & \ddots & \vdots & \vdots & {\alpha'}^{(k)}_{i,j} & \vdots \\
0 & 0 & \alpha^{(k)}_{v,v} & \alpha^{(k)}_{v,v+1} & \dots & \alpha^{(k)}_{v,n} \\
0 & \dots & 0 & 0 & \dots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \dots & 0 & 0 & \dots & 0
\end{pmatrix}
\begin{pmatrix} \tilde{v}_1 \\ \vdots \\ \tilde{v}_v \\ y_1 \\ \vdots \\ y_m \end{pmatrix}
= l^{(k)}_1 \cdot y_1 + \ldots + l^{(k)}_j \cdot y_j + \ldots + c^{(k)}
$$

- One coefficient in the k-th linear equation is altered

Bit-Flip in Central Map

Fault propagation

$$\mathbf{t} \xrightarrow{\mathcal{F}'^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \mathbf{s}'$$

- Faulted signature $\mathbf{s}'$ of $\mathbf{t}$ might deviate heavily from fault-free $\mathbf{s} = T^{-1} \circ \mathcal{F}^{-1}(\mathbf{t})$

Bit-Flip in Central Map

Fault propagation

$$\mathbf{t} \xrightarrow{\mathcal{F}'^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \mathbf{s}'$$

- Faulted signature $\mathbf{s}'$ of $\mathbf{t}$ might deviate heavily from fault-free $\mathbf{s} = T^{-1} \circ \mathcal{F}^{-1}(\mathbf{t})$
- But $\mathcal{P}(\mathbf{s}')$ only deviates in one entry from $\mathbf{t}$

$$\begin{aligned} \mathcal{P}(\mathbf{s}') - \mathbf{t} &= \mathcal{F} \circ T(\mathbf{s}') - \mathcal{F}' \circ T(\mathbf{s}') = (\mathcal{F} - \mathcal{F}') \circ T(\mathbf{s}') \\ &= (0, \dots, 0, (\alpha_{i,j}^{(k)} - \alpha'^{(k)}_{i,j})(T(\mathbf{s}')_i \cdot T(\mathbf{s}')_j), 0, \dots, 0) \end{aligned}$$

Bit-Flip in Central Map

Fault propagation

$$\mathbf{t} \xrightarrow{\mathcal{F}'^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \mathbf{s}'$$

- Faulted signature $\mathbf{s}'$ of $\mathbf{t}$ might deviate heavily from fault-free $\mathbf{s} = T^{-1} \circ \mathcal{F}^{-1}(\mathbf{t})$
- But $\mathcal{P}(\mathbf{s}')$ only deviates in one entry from $\mathbf{t}$

$$\begin{aligned} \mathcal{P}(\mathbf{s}') - \mathbf{t} &= \mathcal{F} \circ T(\mathbf{s}') - \mathcal{F}' \circ T(\mathbf{s}') = (\mathcal{F} - \mathcal{F}') \circ T(\mathbf{s}') \\ &= (0, \ldots, 0, (\alpha_{i,j}^{(k)} - \alpha'^{(k)}_{i,j})(T(\mathbf{s}')_i \cdot T(\mathbf{s}')_j), 0, \ldots, 0) \end{aligned}$$

- This yields quadratic equations in the i-th and j-th row of T

Key Recovery

Iterate the following steps to achieve key recovery (Details in [FKN+22][10])

1. Employ signing oracle to get $N = n(n+1)/2$ message and faulted signature pairs
2. Obtain rows of the secret transformation T
3. Transform $\mathcal{P}$ to a smaller system by reducing the number of variables

[10] Furue, H., Kiyomura, Y., Nagasawa, T., and Takagi, T.: A New Fault Attack on UOV Multivariate Signature Scheme. PQCrypto 2022

Summary of the Fault Attack [FKN+22]

Summary

- Randomization fault
- Attack needs $\approx 10 - 20$ iterations with $n^2/2$ queries to a signing oracle each round

Summary of the Fault Attack [FKN+22]

Summary

- Randomization fault
- Attack needs $\approx 10 - 20$ iterations with $n^2/2$ queries to a signing oracle each round

Practicality

- Purely theoretical $\rightarrow$ No execution of the fault attack yet

Summary of the Fault Attack [FKN+22]

Summary

- Randomization fault
- Attack needs $\approx 10 - 20$ iterations with $n^2/2$ queries to a signing oracle each round

Practicality

- Purely theoretical $\rightarrow$ No execution of the fault attack yet

Countermeasures

- Verify before returning the signature, since faulted signature is invalid
- Check if secret key is altered

Summary of the Fault Attack [FKN+22]

Summary

- Randomization fault
- Attack needs $\approx 10 - 20$ iterations with $n^2/2$ queries to a signing oracle each round

Practicality

- Purely theoretical $\rightarrow$ No execution of the fault attack yet

Countermeasures

- Verify before returning the signature, since faulted signature is invalid
- Check if secret key is altered

Future work

- Find a way to physically cause the randomization in exactly one entry
- Transfer the attack to implementation with compressed keys, where the central map is not stored

A Fault Attack on LUOV [MIS20]

QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme[11]

[11] Mus, K., Islam, S., and Sunar, B. QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme. ACM SIGSAC Conference on Computer and Communications Security 2020

A Fault Attack on LUOV [MIS20]

QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme[11]

- Uses Rowhammer attack to introduce faults to the linear transformation T
  $\rightarrow$ Activate DRAM rows rapidly, to flip bits in neighboring rows (pushes voltage level above or below some threshold)

[11] Mus, K., Islam, S., and Sunar, B. QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme. ACM SIGSAC Conference on Computer and Communications Security 2020

A Fault Attack on LUOV [MIS20]

QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme[11]

- Uses Rowhammer attack to introduce faults to the linear transformation T
  $\rightarrow$ Activate DRAM rows rapidly, to flip bits in neighboring rows (pushes voltage level above or below some threshold)
- Software-induced hardware-fault attack

[11] Mus, K., Islam, S., and Sunar, B. QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme. ACM SIGSAC Conference on Computer and Communications Security 2020

A Fault Attack on LUOV [MIS20]

QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme[11]

- Uses Rowhammer attack to introduce faults to the linear transformation T
  $\rightarrow$ Activate DRAM rows rapidly, to flip bits in neighboring rows (pushes voltage level above or below some threshold)
- Software-induced hardware-fault attack
- Applied the attack with $\approx$ 4hrs of active Rowhammer with efficient post-processing to achieve full key recovery

[11] Mus, K., Islam, S., and Sunar, B. QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme. ACM SIGSAC Conference on Computer and Communications Security 2020

A Fault Attack on LUOV [MIS20]

QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme[11]

- Uses Rowhammer attack to introduce faults to the linear transformation T
  $\rightarrow$ Activate DRAM rows rapidly, to flip bits in neighboring rows (pushes voltage level above or below some threshold)
- Software-induced hardware-fault attack
- Applied the attack with $\approx$ 4hrs of active Rowhammer with efficient post-processing to achieve full key recovery
- Might be transferred to UOV

[11] Mus, K., Islam, S., and Sunar, B. QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme. ACM SIGSAC Conference on Computer and Communications Security 2020

Side Channel Attacks

Horizontal SCA on Linear Transformation T

Main idea[12]

$$\mathrm{t} \xrightarrow{\quad\mathcal{F}^{-1}\quad} \begin{pmatrix} \mathrm{v} \\ \mathrm{y} \end{pmatrix} \xrightarrow{\quad T^{-1}\quad} \begin{pmatrix} \mathrm{s}_1 \\ \mathrm{s}_2 \end{pmatrix} = \begin{pmatrix} \mathrm{v} + T_1 \cdot \mathrm{y} \\ \mathrm{y} \end{pmatrix}$$

- Perform power analysis of matrix-vector multiplication

$$\begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix} \cdot \begin{pmatrix} \mathrm{v} \\ \mathrm{y} \end{pmatrix} = \begin{pmatrix} \mathrm{v} + T_1 \cdot \mathrm{y} \\ \mathrm{y} \end{pmatrix}$$

[12] Park, A., Shim, K. A., Koo, N., and Han, D. G.: Side-channel Attacks on Post-quantum Signature Schemes based on Multivariate Quadratic Equations:-Rainbow and UOV. IACR TCHES 2018

Horizontal SCA on Linear Transformation T

Main idea[12]

$$\mathbf{t} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

- Perform power analysis of matrix-vector multiplication

$$\begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

- Here, the vector $\mathbf{y}$ is known, and the matrix T_1 is the secret we want to obtain

[12] Park, A., Shim, K. A., Koo, N., and Han, D. G.: Side-channel Attacks on Post-quantum Signature Schemes based on Multivariate Quadratic Equations:-Rainbow and UOV. IACR TCHES 2018

Horizontal SCA on Linear Transformation T

Main idea[12]

$$\mathbf{t} \xrightarrow{\mathcal{F}^{-1}} \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} \xrightarrow{T^{-1}} \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

- Perform power analysis of matrix-vector multiplication

$$\begin{pmatrix} I_v & T_1 \\ 0 & I_m \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{v} + T_1 \cdot \mathbf{y} \\ \mathbf{y} \end{pmatrix}$$

- Here, the vector $\mathbf{y}$ is known, and the matrix T_1 is the secret we want to obtain
- Either obtain all entries of T by SCA or identify certain rows and reduce the system $\mathcal{P}$ as shown in previous fault attack

[12] Park, A., Shim, K. A., Koo, N., and Han, D. G.: Side-channel Attacks on Post-quantum Signature Schemes based on Multivariate Quadratic Equations:-Rainbow and UOV. IACR TCHES 2018

Matrix-Vector Multiplication

The vulnerable function in more detail

$$\begin{pmatrix} I & T_1 \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_{1,4} & t_{1,5} \\ 0 & 1 & 0 & t_{2,4} & t_{2,5} \\ 0 & 0 & 1 & t_{3,4} & t_{3,5} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} t_{1,4} \cdot y_1 + t_{1,5} \cdot y_2 \\ t_{2,4} \cdot y_1 + t_{2,5} \cdot y_2 \\ t_{3,4} \cdot y_1 + t_{3,5} \cdot y_2 \\ 0 \\ 0 \end{pmatrix}$$

Matrix-Vector Multiplication

The vulnerable function in more detail

$$\begin{pmatrix} I & T_1 \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_{1,4} & t_{1,5} \\ 0 & 1 & 0 & t_{2,4} & t_{2,5} \\ 0 & 0 & 1 & t_{3,4} & t_{3,5} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} t_{1,4} \cdot y_1 + t_{1,5} \cdot y_2 \\ t_{2,4} \cdot y_1 + t_{2,5} \cdot y_2 \\ t_{3,4} \cdot y_1 + t_{3,5} \cdot y_2 \\ 0 \\ 0 \end{pmatrix}$$

Correlation power analysis

1. Guess intermediate values and map hypothetical value to hypothetical power consumption of the function under investigation

Matrix-Vector Multiplication

The vulnerable function in more detail

$$\begin{pmatrix} I & T_1 \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_{1,4} & t_{1,5} \\ 0 & 1 & 0 & t_{2,4} & t_{2,5} \\ 0 & 0 & 1 & t_{3,4} & t_{3,5} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} t_{1,4} \cdot y_1 + t_{1,5} \cdot y_2 \\ t_{2,4} \cdot y_1 + t_{2,5} \cdot y_2 \\ t_{3,4} \cdot y_1 + t_{3,5} \cdot y_2 \\ 0 \\ 0 \end{pmatrix}$$

Correlation power analysis

1. Guess intermediate values and map hypothetical value to hypothetical power consumption of the function under investigation
2. Measure the power consumption of the target device

Matrix-Vector Multiplication

The vulnerable function in more detail

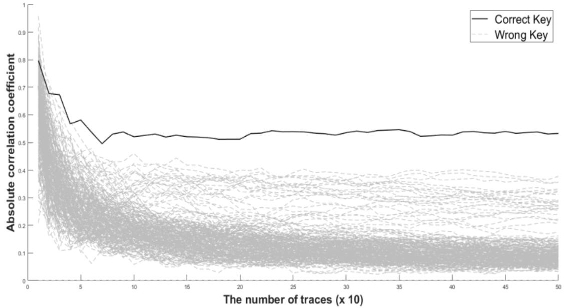
$$\begin{pmatrix} I & T_1 \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_{1,4} & t_{1,5} \\ 0 & 1 & 0 & t_{2,4} & t_{2,5} \\ 0 & 0 & 1 & t_{3,4} & t_{3,5} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} t_{1,4} \cdot y_1 + t_{1,5} \cdot y_2 \\ t_{2,4} \cdot y_1 + t_{2,5} \cdot y_2 \\ t_{3,4} \cdot y_1 + t_{3,5} \cdot y_2 \\ 0 \\ 0 \end{pmatrix}$$

Correlation power analysis

1. Guess intermediate values and map hypothetical value to hypothetical power consumption of the function under investigation
2. Measure the power consumption of the target device
3. Perform statistical comparison between hypothetical power consumption and measured power traces

Compute Correlation with Hypothetical Values

Example with clear separation between correct key elements and wrong key element

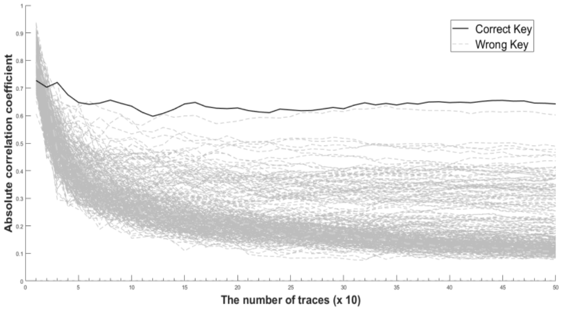


(a) Maximum correlation coefficients according to increased number of traces for $\widetilde{t}_{45}$

Correlation coefficients for all possible field elements and the entry t_{45} [PSK+18]

Compute Correlation with Hypothetical Values

Example with two possible candidate for the correct key element



(b) Maximum correlation coefficients according to increased number of traces for $\widetilde{t}_{46}$

Correlation coefficients for all possible field elements and the entry t_{46} [PSK+18]

Summary of the Horizontal SCA

Summary

- Correlation power analysis on field multiplication
- Around 30 – 100 power traces are needed to recover field elements

Summary of the Horizontal SCA

Summary

- Correlation power analysis on field multiplication
- Around $30 - 100$ power traces are needed to recover field elements

Practicality

- Attack the matrix-vector product code on the ChipWhisperer-Lite evaluation platform
- Target board is an 8-bit Atmel XMEGA128 (might be more difficult on 32-bit devices)
- Parameters were strongly reduced ($n = 8$ and $m = 6$)

Summary of the Horizontal SCA

Summary

- Correlation power analysis on field multiplication
- Around $30 - 100$ power traces are needed to recover field elements

Practicality

- Attack the matrix-vector product code on the ChipWhisperer-Lite evaluation platform
- Target board is an 8-bit Atmel XMEGA128 (might be more difficult on 32-bit devices)
- Parameters were strongly reduced ($n = 8$ and $m = 6$)

Countermeasures

- Masking or shuffling are classical countermeasures for this
- Randomization of the input value (since T is linear)

Summary of the Horizontal SCA

Summary

- Correlation power analysis on field multiplication
- Around $30 - 100$ power traces are needed to recover field elements

Practicality

- Attack the matrix-vector product code on the ChipWhisperer-Lite evaluation platform
- Target board is an 8-bit Atmel XMEGA128 (might be more difficult on 32-bit devices)
- Parameters were strongly reduced ($n = 8$ and $m = 6$)

Countermeasures

- Masking or shuffling are classical countermeasures for this
- Randomization of the input value (since T is linear)

Future work

- Analyze efficiency impact of countermeasures
- Transfer the attack to modern and optimized implementations

Main idea[13]

- Measure power consumption of $\mathcal{P}(\mathbf{v})$

[13]Aulbach, T., Campos, F., Krämer, J., Samardjiska, S., and Stöttinger, M.: Separating Oil and Vinegar with a Single Trace: Side-Channel Assisted Kipnis-Shamir Attack on UOV. IACR TCHES 2023

Attack Insertion of Vinegar Values in Public Key Map

Main idea[13]

- Measure power consumption of $\mathcal{P}(\mathbf{v})$
- This operation boils down to computing $\mathbf{v}^\top P^{(k)}\mathbf{v}$ for m known matrices $P^{(k)}$

[13] Aulbach, T., Campos, F., Krämer, J., Samardjiska, S., and Stöttinger, M.: Separating Oil and Vinegar with a Single Trace: Side-Channel Assisted Kipnis-Shamir Attack on UOV. IACR TCHES 2023

Main idea[13]

- Measure power consumption of $\mathcal{P}(\mathbf{v})$
- This operation boils down to computing $\mathbf{v}^\top P^{(k)}\mathbf{v}$ for m known matrices $P^{(k)}$
- Consider the matrix-vector multiplication

$$P^{(k)} \cdot \mathbf{v} = \begin{pmatrix} p_{1,1}^{(k)} & p_{1,2}^{(k)} & \cdots & p_{1,n}^{(k)} \\ & p_{2,2}^{(k)} & \cdots & p_{2,n}^{(k)} \\ & & \ddots & \vdots \\ & & & p_{n,n}^{(k)} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \text{ for } k \in \{1, \ldots, m\}$$

[13] Aulbach, T., Campos, F., Krämer, J., Samardjiska, S., and Stöttinger, M.: Separating Oil and Vinegar with a Single Trace: Side-Channel Assisted Kipnis-Shamir Attack on UOV. IACR TCHES 2023

Attack Insertion of Vinegar Values in Public Key Map

Main idea[13]

- Measure power consumption of $\mathcal{P}(\mathbf{v})$
- This operation boils down to computing $\mathbf{v}^\top P^{(k)}\mathbf{v}$ for m known matrices $P^{(k)}$
- Consider the matrix-vector multiplication

$$P^{(k)} \cdot \mathbf{v} = \begin{pmatrix} p_{1,1}^{(k)} & p_{1,2}^{(k)} & \cdots & p_{1,n}^{(k)} \\ & p_{2,2}^{(k)} & \cdots & p_{2,n}^{(k)} \\ & & \ddots & \vdots \\ & & & p_{n,n}^{(k)} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \text{ for } k \in \{1, \ldots, m\}$$

- Secret $\mathbf{v}$ is multiplied with a considerable amount of known values

[13] Aulbach, T., Campos, F., Krämer, J., Samardjiska, S., and Stöttinger, M.: Separating Oil and Vinegar with a Single Trace: Side-Channel Assisted Kipnis-Shamir Attack on UOV. IACR TCHES 2023

Template Attack

- Create a template by tracing the power consumption of

$$P^{(k)} \cdot \mathbf{v} = \begin{pmatrix} p_{1,1}^{(k)} & p_{1,2}^{(k)} & \cdots & p_{1,n}^{(k)} \\ & p_{2,2}^{(k)} & \cdots & p_{2,n}^{(k)} \\ & & \ddots & \vdots \\ & & & p_{n,n}^{(k)} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \text{ for } k \in \{1, \ldots, m\}$$

for $\mathbf{v} = \mathbf{0}, \mathbf{1}, \mathbf{2}, \ldots, \mathbf{q-1} \in \mathbb{F}_q^m$

Template Attack

- Create a template by tracing the power consumption of

$$P^{(k)} \cdot \boldsymbol{v} = \begin{pmatrix} p^{(k)}_{1,1} & p^{(k)}_{1,2} & \cdots & p^{(k)}_{1,n} \\ & p^{(k)}_{2,2} & \cdots & p^{(k)}_{2,n} \\ & & \ddots & \vdots \\ & & & p^{(k)}_{n,n} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \text{ for } k \in \{1, \ldots, m\}$$

  for $\boldsymbol{v} = \mathbf{0}, \mathbf{1}, \mathbf{2}, ..., \boldsymbol{q} - \mathbf{1} \in \mathbb{F}_q^m$
- Multiplication of field elements

$$p^{(k)}_{1,1} \cdot v_1 \quad p^{(k)}_{2,2} \cdot v_2 \quad \ldots \quad p^{(k)}_{n,n} \cdot v_n$$

Template Attack

- Create a template by tracing the power consumption of

$$P^{(k)} \cdot \boldsymbol{v} = \begin{pmatrix} p_{1,1}^{(k)} & p_{1,2}^{(k)} & \cdots & p_{1,n}^{(k)} \\ & p_{2,2}^{(k)} & \cdots & p_{2,n}^{(k)} \\ & & \ddots & \vdots \\ & & & p_{n,n}^{(k)} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \text{ for } k \in \{1, \ldots, m\}$$

  for $\boldsymbol{v} = \boldsymbol{0}, \boldsymbol{1}, \boldsymbol{2}, ..., \boldsymbol{q-1} \in \mathbb{F}_q^m$
- Multiplication of field elements

$$p_{1,1}^{(k)} \cdot v_1 \quad p_{2,2}^{(k)} \cdot v_2 \quad \ldots \quad p_{n,n}^{(k)} \cdot v_n$$

- For each field element, we need to run and trace the matrix-vector multiplication only once $\rightarrow$ in total $q = 256$ profiling traces

Template Attack

- Create a template by tracing the power consumption of

$$P^{(k)} \cdot \mathbf{v} = \begin{pmatrix} p_{1,1}^{(k)} & p_{1,2}^{(k)} & \cdots & p_{1,n}^{(k)} \\ & p_{2,2}^{(k)} & \cdots & p_{2,n}^{(k)} \\ & & \ddots & \vdots \\ & & & p_{n,n}^{(k)} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \text{ for } k \in \{1, \dots, m\}$$

  for $\mathbf{v} = \mathbf{0}, \mathbf{1}, \mathbf{2}, ..., \mathbf{q-1} \in \mathbb{F}_q^m$
- Multiplication of field elements

$$p_{1,1}^{(k)} \cdot v_1 \quad p_{2,2}^{(k)} \cdot v_2 \quad \dots \quad p_{n,n}^{(k)} \cdot v_n$$

- For each field element, we need to run and trace the matrix-vector multiplication only once $\rightarrow$ in total $q = 256$ profiling traces
- Can be collected on another device (subtract some mean to erase the 'footprint' of the device)

Record Power Traces

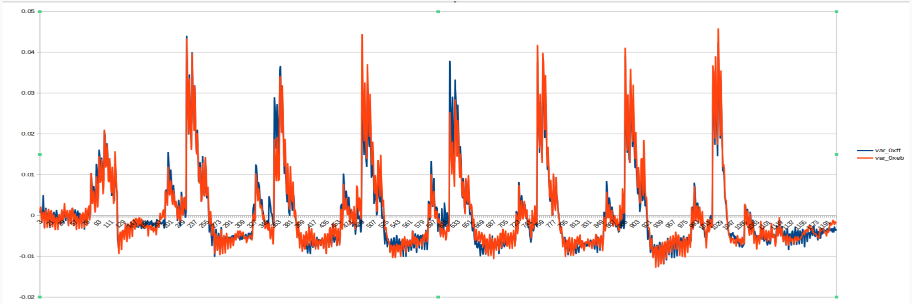
- Power traces are very distinctive

Record Power Traces

- Power traces are very distinctive
- The vinegar variables are processed bitwise from LSB to MSB

Record Power Traces

- Power traces are very distinctive
- The vinegar variables are processed bitwise from LSB to MSB
- Consider the following example



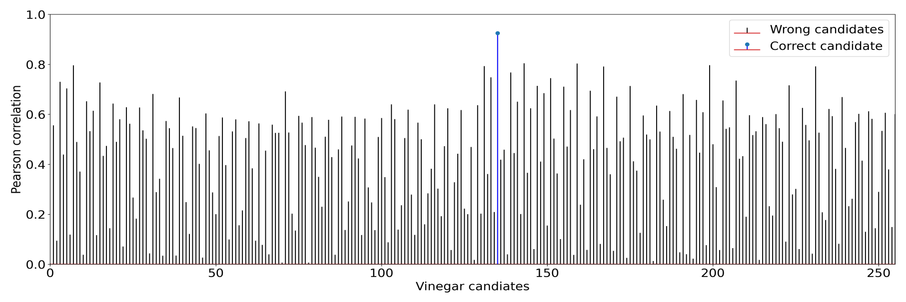
Compare power traces with $v_i = 0xFF$ vs $v_i = 0xEB$

Compute Correlation

- Trace the matrix-vector multiplications with secret vinegar variables on the target device

Compute Correlation

- Trace the matrix-vector multiplications with secret vinegar variables on the target device
- Compute correlation to templates for each entry of $\boldsymbol{v}$



Correlation of the target trace with each of the 256 reference traces

Summary of the Template Attack

Summary

- Very high success probability ($\approx 97\%$) for all vinegar variables
- Template attack with small number of profiling traces
- One single attack trace leads to a secret oil vector (key recovery)

Summary of the Template Attack

Summary

- Very high success probability ($\approx 97\%$) for all vinegar variables
- Template attack with small number of profiling traces
- One single attack trace leads to a secret oil vector (key recovery)

Practicality

- Attack executed with the ChipWhisperer-Lite on an 32-bit STM32F3 target board
- Parameter set only slightly reduced, s.t. $\mathcal{P}$ fits on the target board
- Used modern UOV implementation

Summary of the Template Attack

Summary

- Very high success probability ($\approx 97\%$) for all vinegar variables
- Template attack with small number of profiling traces
- One single attack trace leads to a secret oil vector (key recovery)

Practicality

- Attack executed with the ChipWhisperer-Lite on an 32-bit STM32F3 target board
- Parameter set only slightly reduced, s.t. $\mathcal{P}$ fits on the target board
- Used modern UOV implementation

Countermeasures

- Masking or shuffling are classical countermeasures for this

Summary of the Template Attack

Summary

- Very high success probability ($\approx 97\%$) for all vinegar variables
- Template attack with small number of profiling traces
- One single attack trace leads to a secret oil vector (key recovery)

Practicality

- Attack executed with the ChipWhisperer-Lite on an 32-bit STM32F3 target board
- Parameter set only slightly reduced, s.t. $\mathcal{P}$ fits on the target board
- Used modern UOV implementation

Countermeasures

- Masking or shuffling are classical countermeasures for this

Future work

- Analyze efficiency impact of countermeasures
- Apply the attack to M4 implementations or using a different setup

Takeaways

The End

Takeaways

- Vinegar vectors and oil vectors should be equally secured
- With one of those, the secret key can be recovered in polynomial time
- Some physical attacks are still in a theoretical or simulated state
- Efficiency impact of countermeasures should be analyzed

Questions?
Contact: thomas.aulbach@ur.de

Aulbach, Campos, Krämer, Samardjiska, Stöttinger:
Separating Oil and Vinegar with a Single Trace
https://ia.cr/2023/335

References

- Aulbach, T., Campos, F., Krämer, J., Samardjiska, S., and Stöttinger, M.: *Separating Oil and Vinegar with a Single Trace: Side-Channel Assisted Kipnis-Shamir Attack on UOV*. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2023.
- Aulbach, T., Kovats, T., Krämer, J., and Marzougui, S.: *Recovering Rainbow's Secret Key with a First-Order Fault Attack*. In International Conference on Cryptology in Africa, 2022.
- Beullens, W., Chen, M. S., Hung, S. H., Kannwischer, M. J., Peng, B. Y., Shih, C. J., and Yang, B. Y.: *Oil and Vinegar: Modern Parameters and Implementations*. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2023.
- Furue, H., Kiyomura, Y., Nagasawa, T., and Takagi, T.: *A New Fault Attack on UOV Multivariate Signature Scheme*. In International Conference on Post-Quantum Cryptography, 2022.
- Hashimoto, Y., Takagi, T., and Sakurai, K.: *General Fault Attacks on Multivariate Public Key Cryptosystems*. In International Conference on Post-Quantum Cryptography, 2011.

References

- Kato, T., Kiyomura, Y., and Takagi, T.: *Improving Fault Attacks on Rainbow with Fixing Random Vinegar Values*. International Workshop on Security, 2022.
- Krämer, J., and Loiero, M.: *Fault Attacks on UOV and Rainbow*. In Constructive Side-Channel Analysis and Secure Design: 10th International Workshop, COSADE, 2019.
- Mus, K., Islam, S., and Sunar, B.: *QuantumHammer: a Practical Hybrid Attack on the LUOV Signature Scheme*. In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, 2020.
- Park, A., Shim, K. A., Koo, N., and Han, D. G.: *Side-channel Attacks on Post-quantum Signature Schemes based on Multivariate Quadratic Equations:-Rainbow and UOV*. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2018.
- Shim, K. A., and Koo, N.: *Algebraic Fault Analysis of UOV and Rainbow with the Leakage of Random Vinegar Values*. IEEE Transactions on Information Forensics and Security, 2020.