A template attack against Verify PIN algorithms

Hélène Le Bouder, Thierno Barry, Damien Couroussé, Jean-Louis Lanet and Ronan Lashermes
Personal Identification Number (PIN) codes.

- Used to authenticate the user,
- in payment cards or SIM cards...
- Targets of choice for malicious adversaries.
- A limited number of trials.
Personal Identification Number (PIN) codes.

- Used to authenticate the user,
- in payment cards or SIM cards...
- Targets of choice for malicious adversaries.
- A limited number of trials.

Side Channel Analysis (SCA)

- SCA consists in observing some physical characteristics which are modified during the computation performed on the circuit.
- Most classic leakages are: timing, power consumption, electromagnetic emissions (EM) ...
- The main difficulty of the attack is to succeed with very few traces.
- Template attack is a kind of SCA, based on characterization.
1 Introduction

2 Verify PIN algorithm

3 Attack
   - Profiling phase
   - Attack phase

4 Results
   - Test bench
   - General results
   - Final attack

5 Conclusion
1 Introduction

2 Verify PIN algorithm

3 Attack
   - Profiling phase
   - Attack phase

4 Results
   - Test bench
   - General results
   - Final attack

5 Conclusion
Verify PIN algorithm

1: procedure VERIFY PIN(candidate PIN $V$)
2: \hspace{1em} counter = counter − 1
3: \hspace{1em} if counter > 0 then
4: \hspace{2em} status = COMPARISON($U$, $V$)
5: \hspace{2em} status$_2$ = COMPARISON($U$, $V$)
6: \hspace{2em} if status $\neq$ status$_2$ then
7: \hspace{3em} ERROR, device is blocked
8: \hspace{2em} else
9: \hspace{3em} if status = TRUE then
10: \hspace{4em} counter initialized at original value.
11: \hspace{3em} end if
12: \hspace{2em} end if
13: \hspace{2em} else
14: \hspace{3em} device is blocked
15: \hspace{2em} end if
16: return status
17: end procedure

- PIN code is an array of $m$ bytes.
- **True PIN**: $U$,
- **Candidate PIN**: $V$,
- $U \in [0, 9]^m$.
- $10^m$ different PIN codes.
- Countermeasure against fault attack: compare $U$ and $V$ twice.
Comparison of two PIN codes

1: procedure COMPARISON(candidate PIN $V$, true PIN $U$)
2: \hspace{1em} status = FALSE
3: \hspace{1em} diff = FALSE
4: \hspace{1em} fake = FALSE
5: \hspace{1em} for $b = 0$ to $m$ do
6: \hspace{2em} if $U_b \neq V_b$ then
7: \hspace{3em} diff = TRUE
8: \hspace{2em} else
9: \hspace{3em} fake = TRUE
10: \hspace{2em} end if
11: \hspace{1em} if ($b = m$) and (diff = FALSE) then
12: \hspace{2em} status = TRUE
13: \hspace{1em} else
14: \hspace{2em} fake = TRUE
15: \hspace{1em} end if
16: \hspace{1em} end for
17: return status
18: end procedure

Countermeasure against timing attack: comparison between $U$ and $V$ has to be in a constant time.
1. Introduction

2. Verify PIN algorithm

3. Attack
   - Profiling phase
   - Attack phase

4. Results
   - Test bench
   - General results
   - Final attack

5. Conclusion
A template attack

2 phases

1. profiling phase,
2. attack phase.

The attacker can:
- obtain one trace on the targeted device;
- change the True PIN in her profiling device;
- obtain many traces on her profiling device.
1 Introduction

2 Verify PIN algorithm

3 Attack
   - Profiling phase
   - Attack phase

4 Results
   - Test bench
   - General results
   - Final attack

5 Conclusion
Profiling phase

On the profiling device

**Step 1: Campaign on the profiling device**

- Campaign is for one given byte $b$.
- The byte $U_b$ of the True PIN takes all values $k$ in $[0, 9]$ and the other bytes stay to zero.
- Bytes of Candidate PIN $V$ are fixed to a chosen value $v$.
- For each $(k, v)$ collect many traces: $M_{v,k} = \{x_k(i, j)\}$, $i$ for trace, $j$ for time.
On the profiling device

Step 1: Campaign on the profiling device

- Campaign is for one given byte $b$.
- The byte $U_b$ of the True PIN takes all values $k$ in $[0, 9]$ and the other bytes stay to zero.
- Bytes of Candidate PIN $V$ are fixed to a chosen value $v$.
- For each $(k, v)$ collect many traces: $M_{v,k} = \{x_{k(i,j)}\}$, $i$ for trace, $j$ for time.

Step 2: Detection of points of interest.

Select the moment of computation of Comparison (relevant $j$).
On the profiling device

**Step 1: Campaign on the profiling device**
- Campaign is for one given byte $b$.
- The byte $U_b$ of the True PIN takes all values $k$ in $[0, 9]$ and the other bytes stay to zero.
- Bytes of Candidate PIN $V$ are fixed to a chosen value $v$.
- For each $(k, v)$ collect many traces: $M_{v,k} = \{x_k(i,j)\}$, $i$ for trace, $j$ for time.

**Step 2: Detection of points of interest.**
Select the moment of computation of Comparison (relevant $j$).

**Step 3: Build of templates.**
- Compute the sample covariance matrix $S_{v,k} = \{sk(j,j')\}$,
  
  $sk(j,j') = \frac{1}{n-1} \cdot (x_{kj} - \bar{x}_{kj})^t (x_{kj'} - \bar{x}_{kj'})$.

A template attack against Verify PIN algorithms

Le Boudier et al.

July 27th 2016
1. Introduction

2. Verify PIN algorithm

3. Attack
   - Profiling phase
   - Attack phase

4. Results
   - Test bench
   - General results
   - Final attack

5. Conclusion
Step 4: Campaign on the targeted device

- True PIN byte $U_b$ is unknown, it is the target;
- Candidate PIN byte $V_b$ is equal to $v$.
- Trace is a vector $T_v = \{x_j\}$. 
On targeted device

Step 4: Campaign on the targeted device
- True PIN byte $U_b$ is unknown, it is the target;
- Candidate PIN byte $V_b$ is equal to $v$.
- Trace is a vector $T_v = \{x_j\}$.

Step 5: Confrontation between measurements
- Confront the trace $T_v$ to the template matrix $S_{v,k}$.
- General formula in template attack:
  $$F_v(T_v|S_{v,k}, \bar{x}k) = \frac{1}{\sqrt{(2\pi)^p.|S_{v,k}|}} \cdot \exp\left( -\frac{1}{2} \cdot (T_v - \bar{x}k) \cdot S_{v,k}^{-1} \cdot (T_v - \bar{x}k)^t \right).$$
On targeted device

**Step 4: Campaign on the targeted device**
- True PIN byte $U_b$ is **unknown**, it is the target;
- Candidate PIN byte $V_b$ is equal to $v$.
- Trace is a vector $T_v = \{x_j\}$.

**Step 5: Confrontation between measurements**
- Confront the trace $T_v$ to the template matrix $S_{v,k}$.
- General formula in template attack:
  \[
  F_v(T_v|S_{v,k}, x_k) = \frac{1}{\sqrt{(2\pi)^p \cdot |S_{v,k}|}} \cdot \exp \left( -\frac{1}{2} \cdot (T_v - x_k) \cdot S_{v,k}^{-1} \cdot (T_v - x_k)^t \right).
  \]

**Step 6: Discriminating guesses**
- Return the guess $k_v$ for which $F_v$ is maximal for a given $T_v$.
- Rank the guesses $k$ according to the value of $F_v(T_v, k)$. 
<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
<th>Verify PIN algorithm</th>
<th>Attack</th>
<th>Results</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Verify PIN algorithm</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Profiling phase</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Attack phase</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Results</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Test bench</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>General results</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Final attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Conclusion</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
A template attack against Verify PIN algorithms

Le Bouter et al.

July 27th 2016
1 Introduction

2 Verify PIN algorithm

3 Attack
   • Profiling phase
   • Attack phase

4 Results
   • Test bench
   • General results
   • Final attack

5 Conclusion
The True byte PIN: $U_b = 0$

The Candidate byte PIN: $V_b = 0$

The returned guess is clearly: $k = 0$

If $U_b = V_b$, The attack always succeeds.
The True PIN byte: \( U_b = 3 \).

The Candidate PIN byte: \( V_b = 0 \).

The returned guess is \( k = 3 \).

\( U_b \neq V_b \): The attack succeeds, not so clearly.
| 1 | Introduction |
| 2 | Verify PIN algorithm |
| 3 | Attack |
|   | • Profiling phase |
|   | • Attack phase |
| 4 | Results |
|   | • Test bench |
|   | • General results |
|   | • Final attack |
| 5 | Conclusion |
1: **procedure** ATTACK\((C\) the number of trials in the VERFY PIN)\)

2: \(N = C - 1\) // limitation of number trials.

3: \(v = 0\)

4: \(K = [0, 9]\)

5: \(\hat{k} = \max_{k \in K} (F_v(T_v, k))\) // \(\hat{k}\) best guess with \(v\).

6: **while** \(\hat{k} \neq v\) and \(N > 0\) **do**

7: \(N = N - 1\)

8: \(K = K \setminus \{v\}\) // guess \(v\) is eliminated.

9: \(v = \hat{k}\)

10: \(\hat{k} = \max_{k \in K}^{-1} (F_v(T_v, k))\).

11: **end while**

12: return \(\hat{k}\)

13: **end procedure**

- \(v\) is the value tested on the Candidate PIN: \(V_b = v\).

- \(F_v(T_v, k)\) function template of the attack.

1. Send candidate PIN with all bytes to 0.

2. Then test the PIN code returned by the first attack.

- **Worst case:** in 8 trials, the PIN code is retrieved.
Success rate

<table>
<thead>
<tr>
<th>number of traces:</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n = 100000 )</td>
<td>1 COMPARAISON</td>
<td>27.70</td>
<td>41.47</td>
<td>53.84</td>
<td>63.99</td>
<td>73.07</td>
<td>81.33</td>
<td>88.51</td>
</tr>
<tr>
<td></td>
<td>2 COMPARAISON</td>
<td>31.71</td>
<td>46.56</td>
<td>57.82</td>
<td>67.76</td>
<td>76.63</td>
<td>84.36</td>
<td>90.68</td>
</tr>
<tr>
<td>( n = 200000 )</td>
<td>1 COMPARAISON</td>
<td>29.28</td>
<td>44.27</td>
<td>56.79</td>
<td>67.41</td>
<td>76.66</td>
<td>83.91</td>
<td>90.68</td>
</tr>
<tr>
<td></td>
<td>2 COMPARAISON</td>
<td>32.72</td>
<td>49.52</td>
<td>61.96</td>
<td>72.05</td>
<td>80.49</td>
<td>87.53</td>
<td>93.23</td>
</tr>
<tr>
<td>( n = 400000 )</td>
<td>1 COMPARAISON</td>
<td>29.56</td>
<td>44.11</td>
<td>56.0</td>
<td>66.88</td>
<td>75.96</td>
<td>84.04</td>
<td>90.58</td>
</tr>
<tr>
<td></td>
<td>2 COMPARAISON</td>
<td>32.91</td>
<td>48.38</td>
<td>60.88</td>
<td>71.68</td>
<td>80.07</td>
<td>86.91</td>
<td>92.94</td>
</tr>
</tbody>
</table>

**Success rate to retrieve a byte of a True PIN** \( U_b \) **according to the size** \( n \) **of the templates and the number and the choice of traces.**
1 Introduction

2 Verify PIN algorithm

3 Attack
   - Profiling phase
   - Attack phase

4 Results
   - Test bench
   - General results
   - Final attack

5 Conclusion
The first SCA attack with EM traces on Verify PIN algorithms.

To enter a PIN code, a user has a limited number of trials.

Therefore the main difficulty of the attack is to succeed with very few traces.

The PIN is retrieved in 8 trials at most!
The first SCA attack with EM traces on Verify PIN algorithms.

To enter a PIN code, a user has a limited number of trials.

Therefore the main difficulty of the attack is to succeed with very few traces.

The PIN is retrieved in 8 trials at most!

It becomes a new real threat, and it is feasible on a low cost and portable platform.

Some protections against fault attacks introduce new vulnerabilities.
The first SCA attack with EM traces on Verify PIN algorithms.

To enter a PIN code, a user has a limited number of trials.

Therefore the main difficulty of the attack is to succeed with very few traces.

The PIN is retrieved in 8 trials at most!

It becomes a new real threat, and it is feasible on a low cost and portable platform.

Some protections against fault attacks introduce new vulnerabilities.

Future works:
- Find new countermeasures.
- Test the attack on a real device (mobile phone or smart card).
Thank you for your attention!

Any questions?