oberthur TECHNOLOGIES
THE M COMPANY

Lost in Translation

Fault Analysis of Infective Security Proofs

Alberto Battistello and Christophe Giraud(y)

Oberthur Technologies
Security Group
{a.battistello,c.giraud}@oberthur.com

September 11, 2015

oberthur TECHNOLOGIES THE M COMPANY

Overview

1. Introduction
2. FDTC 14 Infective Countermeasure Description
3. Fault Analysis
4. Conclusion

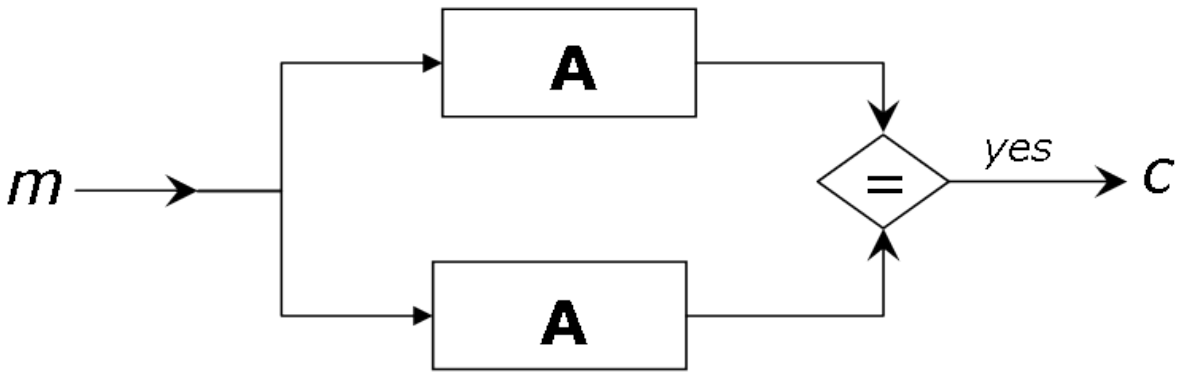
oberthur TECHNOLOGIES THE M COMPANY

Overview

oberthur TECHNOLOGIES THE M COMPANY

Countermeasures
Detection

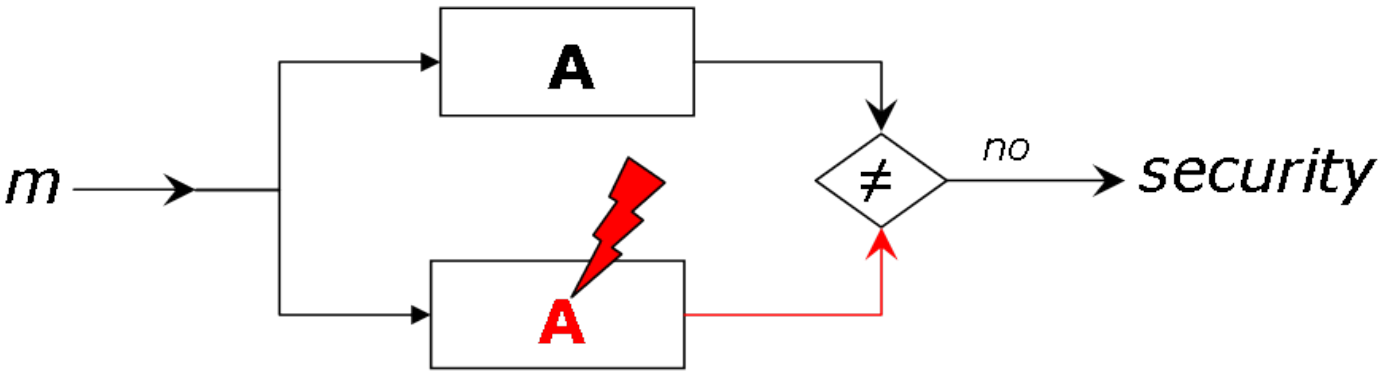
Detection Countermeasure



oberthur TECHNOLOGIES THE M COMPANY

Countermeasures
Detection

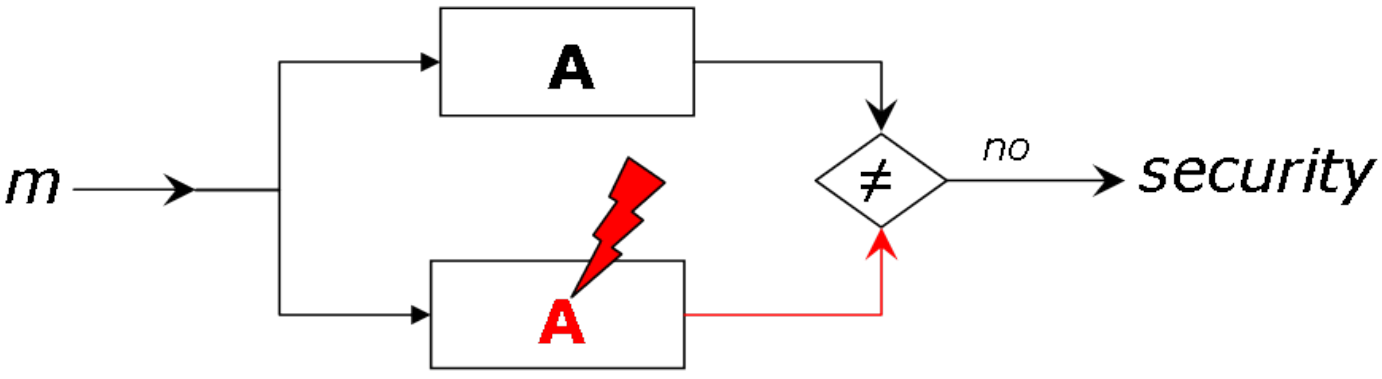
Detection Countermeasure



oberthur TECHNOLOGIES THE M COMPANY

Countermeasures
Detection

Detection Countermeasure



Drawbacks

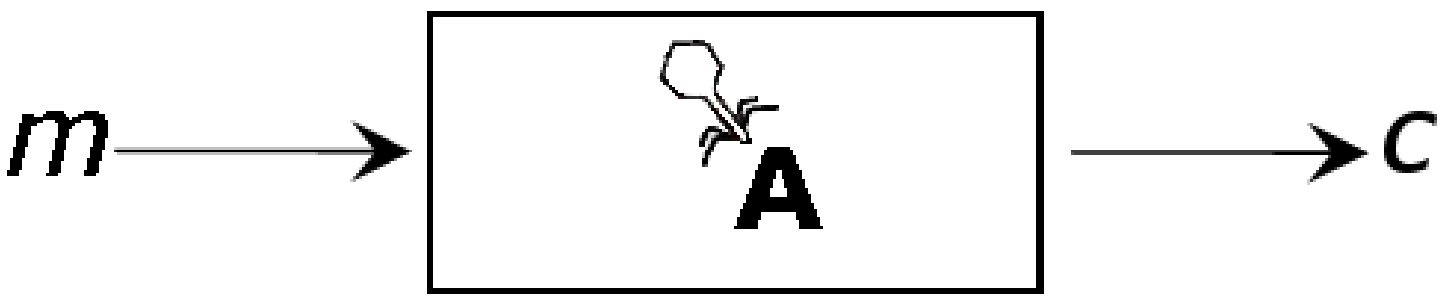
Comparison

Different paths

oberthur TECHNOLOGIES THE M COMPANY

Countermeasures
Infection

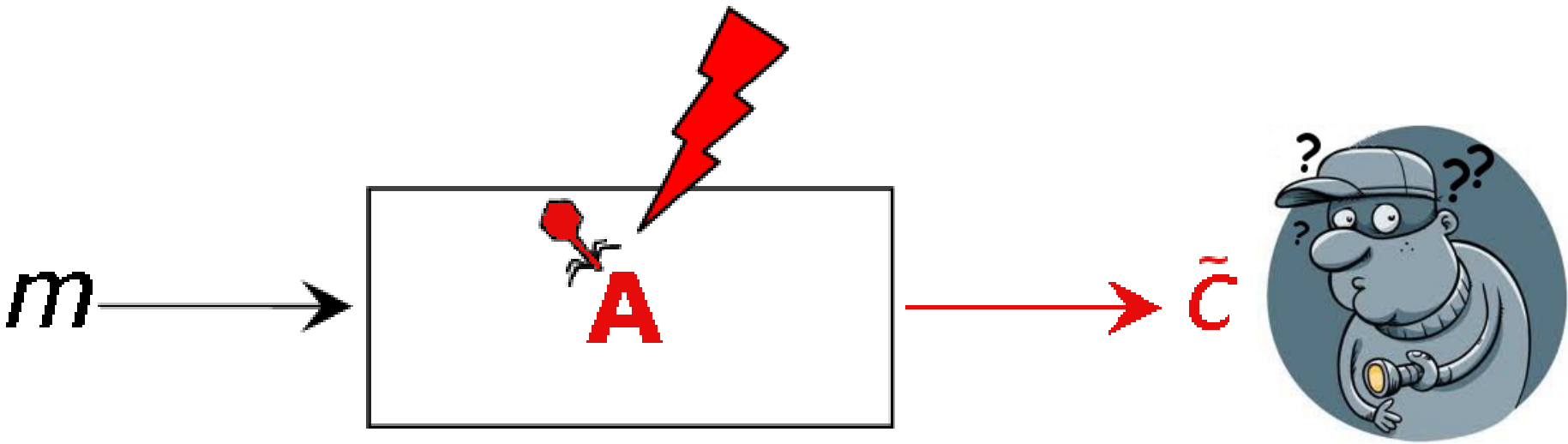
Infection Method



oberthur TECHNOLOGIES THE M COMPANY

Countermeasures
Infection

Infection Method



oberthur TECHNOLOGIES THE M COMPANY

Asymmetric Infections

1. [Yen, Kim, Lim, Moon] 2001 $\Rightarrow$ [Yen, Kim, Moon] 2004
2. [Blömer, Otto, Seifert] 2003 $\Rightarrow$ [Wagner] 2004
3. [Ciet, Joye] 2005 $\Rightarrow$ [Berzati, Canovas, Goubin] 2008
4. [Schmidt *et al.*] 2010 $\Rightarrow$ [Feix, Venelli] 2013

oberthur TECHNOLOGIES THE M COMPANY

Asymmetric Infections

1. [Yen, Kim, Lim, Moon] 2001 $\Rightarrow$ [Yen, Kim, Moon] 2004
2. [Blömer, Otto, Seifert] 2003 $\Rightarrow$ [Wagner] 2004
3. [Ciet, Joye] 2005 $\Rightarrow$ [Berzati, Canovas, Goubin] 2008
4. [Schmidt *et al.*] 2010 $\Rightarrow$ [Feix, Venelli] 2013
5. [Guilley, Rauzy] 2014

oberthur TECHNOLOGIES THE M COMPANY

Asymmetric Infections

1. [Yen, Kim, Lim, Moon] 2001 $\Rightarrow$ [Yen, Kim, Moon] 2004
2. [Blömer, Otto, Seifert] 2003 $\Rightarrow$ [Wagner] 2004
3. [Ciet, Joye] 2005 $\Rightarrow$ [Berzati, Canovas, Goubin] 2008
4. [Schmidt *et al.*] 2010 $\Rightarrow$ [Feix, Venelli] 2013
5. [Guilley, Rauzy] 2014 $\Rightarrow$ This Work

oberthur TECHNOLOGIES THE M COMPANY

Overview

oberthur TECHNOLOGIES THE M COMPANY

Translation

Issue with formal analysis:

- Detective countermeasure ☺
- Infective countermeasures ☹

Idea

1. Provide a security certificate for a detective algorithm
2. Translate detection to infection
3. Inherit the certificate.

oberthur TECHNOLOGIES THE M COMPANY

Security Proof

Proposition 2

Each test-based countermeasure has a direct equivalent infective countermeasure.

Proof

Transform:

if a != b mod m then return error

Into:

$c^* = a - b + 1 \bmod m$

oberthur TECHNOLOGIES THE M COMPANY

Security Proof

Proposition 2

Each test-based countermeasure has a direct equivalent infective countermeasure.

Proof

Transform:

```
if a != b mod m then return error
```

Into:

```
c* = a - b + 1 mod m
```

output $\widehat{S} = S^{c^*}$ $\Rightarrow gcd(N, S - \widehat{S}) \neq p, q$

oberthur TECHNOLOGIES THE M COMPANY

Security Proof

Proposition 2

Each test-based countermeasure has a direct equivalent infective countermeasure.

Proof

Transform:

```
if a != b mod m then return error
```

Into:

```
c* = a - b + 1 mod m
```

output $\widehat{S} = S^{c^*}$ $\Rightarrow gcd(N, S - \widehat{S}) \neq p, q$

- Aumüller detective $\Rightarrow$ Aumüller infective
- Vigilant detective $\Rightarrow$ Vigilant infective

oberthur TECHNOLOGIES THE M COMPANY

Overview

oberthur TECHNOLOGIES THE M COMPANY

This Paper

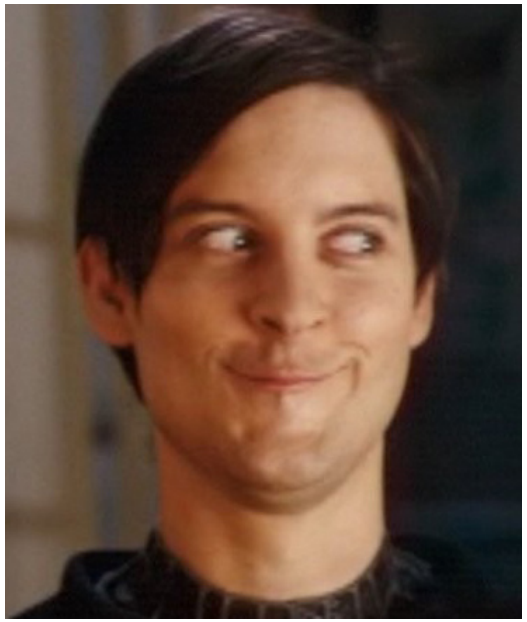
Objection

The security proof is **only** valid vs ”classical” BellCore, i.e:

$$gcd(N, S - S^{c^*})$$

However it may be insecure vs BellCore variants, i.e:

$$gcd(N, S^{c^*} - m^{\gamma})$$



Let’s try it!

oberthur TECHNOLOGIES THE M COMPANY

Fault Model

Definition 1 (Fault injection)**.** During the execution of an algorithm, the attacker can:

- modify any intermediate value by setting it to either a *random value* (*randomizing fault*) or *zero* (*zeroing fault*); such a fault can be either *permanent* (*e.g.*, in memory) or *transient* (*e.g.*, in a register or a bus);
- skip any number of consecutive instructions (*skipping fault*).

Lemma 1. *The effect of a skipping fault (i.e., fault on the code) can be captured by considering only randomizing and zeroing faults (i.e., fault on the data).*

Proof. Indeed, if the skipped instructions are part of an arithmetic operation:

- either the computation has not been done at all and the value in memory where the result is supposed to be stays zero (if initialized) or random (if not),
- or the computation has partly been done and the value written in memory as its result is thus pseudo-randomized (and considered random at our modeling level).

If the skipped instruction is a branching instruction, then it is equivalent to do a zeroing fault on the result of the branching condition to make it false and thus avoid branching. □

We apply

- Randomizing fault
- Stuck-at-0 fault

oberthur TECHNOLOGIES THE M COMPANY

Detective Aumüller

Algorithm 7: CRT-RSA with Aumüller *et al.*'s countermeasure[6] [22]

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or error

1 Choose a small random integer r.

2 $p' = p \cdot r$
3 $q' = q \cdot r$
4 **if** $p' \not\equiv 0 \mod p$ **or** $q' \not\equiv 0 \mod q$ **then return** error

5 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
6 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$

7 $S_p = S'_p \mod p$
8 $S_q = S'_q \mod q$

9 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
10 **if** $S \not\equiv S'_p \mod p$ **or** $S \not\equiv S'_q \mod q$ **then return** error

11 $S_{pr} = S'_p \mod r$
12 $S_{qr} = S'_q \mod r$
13 **if** ${S_{pr}}^{d_q \mod \varphi(r)} \not\equiv {S_{qr}}^{d_p \mod \varphi(r)} \mod r$ **then return** error

14 **return** S

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.

2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$

4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$

// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$

// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$

// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$

13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = {S_{pr}}^{d_q \mod \varphi(r)} - {S_{qr}}^{d_p \mod \varphi(r)} + 1 \mod r$

16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = {S_{pr}}^{d_q \mod \varphi(r)} - {S_{qr}}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault
Message $M = 1$:

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault
Message $M = 1$:

8 $\tilde{S}_p = \epsilon$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault
Message $M = 1$:

8 $\tilde{S}_p = \epsilon$

10

$$\begin{cases} \tilde{S} \equiv S_q = 1 \mod q \\ \tilde{S} \equiv \tilde{S}_p = \epsilon \mod p \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault
Message $M = 1$:

8 $\tilde{S}_p = \epsilon$

10
$$\begin{cases} \tilde{S} \equiv S_q = 1 \bmod q \\ \tilde{S} \equiv \tilde{S}_p = \epsilon \bmod p \end{cases}$$

16 $\forall c_1 \ldots c_5$

$$\begin{cases} \tilde{S}^{c_1 \cdots c_5} \equiv 1^{c_1 \cdots c_5} \equiv 1 \bmod q \\ \tilde{S}^{c_1 \cdots c_5} \equiv \epsilon^{c_1 \cdots c_5} \not\equiv 1 \bmod p \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault
Message $M = 1$:

8 $\tilde{S}_p = \epsilon$

10
$$\begin{cases} \tilde{S} \equiv S_q = 1 \bmod q \\ \tilde{S} \equiv \tilde{S}_p = \epsilon \bmod p \end{cases}$$

16 $\forall c_1 \ldots c_5$

$$\begin{cases} \tilde{S}^{c_1 \cdots c_5} \equiv 1^{c_1 \cdots c_5} \equiv 1 \bmod q \\ \tilde{S}^{c_1 \cdots c_5} \equiv \epsilon^{c_1 \cdots c_5} \not\equiv 1 \bmod p \end{cases}$$

$\Rightarrow \gcd((\tilde{S}^{c_1 \cdots c_5}) - 1, N) = q$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 1

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Random Fault
Message $M = 1$:

8 $\tilde{S}_p = \epsilon$

10
$$\begin{cases} \tilde{S} \equiv S_q = 1 \bmod q \\ \tilde{S} \equiv \tilde{S}_p = \epsilon \bmod p \end{cases}$$

16 $\forall c_1 \ldots c_5$

$$\begin{cases} \tilde{S}^{c_1 \cdots c_5} \equiv 1^{c_1 \cdots c_5} \equiv 1 \bmod q \\ \tilde{S}^{c_1 \cdots c_5} \equiv \epsilon^{c_1 \cdots c_5} \not\equiv 1 \bmod p \end{cases}$$

$\Rightarrow \gcd((\tilde{S}^{c_1 \cdots c_5}) - 1, N) = q$

Available fault targets:

- Steps 6, 7, 8 or 9

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.

2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$

4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$

// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$

// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$

// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$

13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = {S_{pr}}^{d_q \mod \varphi(r)} - {S_{qr}}^{d_p \mod \varphi(r)} + 1 \mod r$

16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 2

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = {S_{pr}}^{d_q \mod \varphi(r)} - {S_{qr}}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Stuck-at-0:

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 2

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Stuck-at-0:

8 $\tilde{S}_p = 0$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 2

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Stuck-at-0:

8 $\tilde{S}_p = 0$

10

$$\begin{cases} \tilde{S} \equiv S_q = {m_q}^{d_q} \mod q \\ \tilde{S} \equiv \tilde{S}_p = 0 \mod p \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 2

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Stuck-at-0:

8 $\tilde{S}_p = 0$

10
$$\begin{cases} \tilde{S} \equiv S_q = {m_q}^{d_q} \mod q \\ \tilde{S} \equiv \tilde{S}_p = 0 \mod p \end{cases}$$

16 $\forall c_1 \ldots c_5$

$$\begin{cases} \tilde{S}^{c_1 \cdots c_5} \equiv {S_q}^{c_1 \cdots c_5} \not\equiv 0 \mod q \\ \tilde{S}^{c_1 \cdots c_5} \equiv 0^{c_1 \cdots c_5} \equiv 0 \mod p \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 2

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Stuck-at-0:

8 $\tilde{S}_p = 0$

10
$$\begin{cases} \tilde{S} \equiv S_q = {m_q}^{d_q} \mod q \\ \tilde{S} \equiv \tilde{S}_p = 0 \mod p \end{cases}$$

16 $\forall c_1 \ldots c_5$

$$\begin{cases} \tilde{S}^{c_1 \cdots c_5} \equiv {S_q}^{c_1 \cdots c_5} \not\equiv 0 \mod q \\ \tilde{S}^{c_1 \cdots c_5} \equiv 0^{c_1 \cdots c_5} \equiv 0 \mod p \end{cases}$$

$\Rightarrow$ $\gcd(\tilde{S}^{c_1 \cdots c_5}, N) = p$

oberthur TECHNOLOGIES THE M COMPANY

Translated Aumüller: Attack 2

Algorithm 13: CRT-RSA with Aumüller *et al.*'s countermeasure[6], under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value

1 Choose a small random integer r.
2 $p' = p \cdot r$
3 $c_1 = p' + 1 \mod p$
4 $q' = q \cdot r$
5 $c_2 = q' + 1 \mod q$
// Intermediate signature in $\mathbb{Z}_{pr}$
6 $S'_p = M^{d_p \mod \varphi(p')} \mod p'$
// Intermediate signature in $\mathbb{Z}_{qr}$
7 $S'_q = M^{d_q \mod \varphi(q')} \mod q'$
// Retrieve intermediate signature in $\mathbb{Z}_p$
8 $S_p = S'_p \mod p$
// Retrieve intermediate signature in $\mathbb{Z}_q$
9 $S_q = S'_q \mod q$
// Recombination in $\mathbb{Z}_N$
10 $S = S_q + q \cdot (i_q \cdot (S_p - S_q) \mod p)$
11 $c_3 = S - S'_p + 1 \mod p$
12 $c_4 = S - S'_q + 1 \mod q$
13 $S_{pr} = S'_p \mod r$ // Checksum of S_p in $\mathbb{Z}_r$
14 $S_{qr} = S'_q \mod r$ // Checksum of S_q in $\mathbb{Z}_r$
15 $c_5 = S_{pr}^{d_q \mod \varphi(r)} - S_{qr}^{d_p \mod \varphi(r)} + 1 \mod r$
16 **return** $S^{c_1 c_2 c_3 c_4 c_5}$

Stuck-at-0:

8 $\tilde{S}_p = 0$

10
$$\begin{cases} \tilde{S} \equiv S_q = {m_q}^{d_q} \mod q \\ \tilde{S} \equiv \tilde{S}_p = 0 \mod p \end{cases}$$

16 $\forall c_1 \ldots c_5$

$$\begin{cases} \tilde{S}^{c_1 \cdots c_5} \equiv {S_q}^{c_1 \cdots c_5} \not\equiv 0 \mod q \\ \tilde{S}^{c_1 \cdots c_5} \equiv 0^{c_1 \cdots c_5} \equiv 0 \mod p \end{cases}$$

$\Rightarrow \gcd(\tilde{S}^{c_1 \cdots c_5}, N) = p$

Available fault targets:

- Steps 6, 7, 8 or 9

oberthur TECHNOLOGIES THE M COMPANY

Vigilant detective

Algorithm 8: CRT-RSA with Vigilant's countermeasure[6] [3]
with Coron *et al.*'s fixes [4] and Rauzy & Guilley's simplifications [5]

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or error

1 Choose small random integers r, R_1, and R_2.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$ // CRT insertion of verification value in M'_p
9 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$ // Intermediate signature in $\mathbb{Z}_{pr^2}$
10 **if** $M'_p \not\equiv M \mod p$ **then return** error
11 **if** $B_p \cdot S'_p \not\equiv B_p \cdot (1 + d_p \cdot r) \mod p'$ **then return** error
12 $q' = q \cdot r^2$
13 $i_{qr} = q^{-1} \mod r^2$
14 $M_q = M \mod q'$
15 $B_q = q \cdot i_{qr}$
16 $A_q = 1 - B_q \mod q'$
17 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$ // CRT insertion of verification value in M'_q
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$ // Intermediate signature in $\mathbb{Z}_{qr^2}$
19 **if** $M'_q \not\equiv M \mod q$ **then return** error
20 **if** $B_q \cdot S'_q \not\equiv B_q \cdot (1 + d_q \cdot r) \mod q'$ **then return** error
21 $S_{pr} = S'_p - B_p \cdot (1 + d_p \cdot r - R_1)$ // Verification value of S'_p swapped with R_1
22 $S_{qr} = S'_q - B_q \cdot (1 + d_q \cdot r - R_2)$ // Verification value of S'_q swapped with R_2
23 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$ // Recombination in $\mathbb{Z}_{Nr^2}$
// Simultaneous verification of lines 2 and 23
24 **if** $pq \cdot (S_r - R_2 - q \cdot i_q \cdot (R_1 - R_2)) \not\equiv 0 \mod Nr^2$ **then return** error
25 **return** $S = S_r \mod N$ // Retrieve result in $\mathbb{Z}_N$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

Choose a small random integer r.
$N = p \cdot q$
$p' = p \cdot r^2$
$i_{pr} = p^{-1} \mod r^2$
$M_p = M \mod p'$
$B_p = p \cdot i_{pr}$
$A_p = 1 - B_p \mod p'$
$M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
$q' = q \cdot r^2$
$i_{qr} = q^{-1} \mod r^2$
$M_q = M \mod q'$
$B_q = q \cdot i_{qr}$
$A_q = 1 - B_q \mod q'$
$M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
$S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
$S_{pr} = 1 + d_p \cdot r$
$c_p = M'_p + N - M + 1 \mod p$
$S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
$S_{qr} = 1 + d_q \cdot r$
$c_q = M'_q + N - M + 1 \mod q$
$S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
$S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
$c_S = S - S_r + 1 \mod r^2$
return $S = S'^{c_p c_q c_S} \mod N$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

- $M = 1$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

- $M = 1$

15 $\tilde{S}'_p \leftarrow \epsilon \mod p'$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

- $M = 1$

15 $\tilde{S}'_p \leftarrow \epsilon \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q \equiv 1 \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p \not\equiv 1 \mod p' \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

- $M = 1$

15 $\tilde{S}'_p \leftarrow \epsilon \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q \equiv 1 \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p \not\equiv 1 \mod p' \end{cases}$$

24 $\forall c_p, c_q, c_S$

$$\begin{cases} \tilde{S} \equiv \tilde{S}'^{c_p c_q c_S}_q \equiv 1^{c_p c_q c_S} \equiv 1 \mod q' \\ \tilde{S} \equiv \tilde{S}'^{c_p c_q c_S}_p \not\equiv 1 \mod p' \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

- $M = 1$

15 $\tilde{S}'_p \leftarrow \epsilon \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q \equiv 1 \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p \not\equiv 1 \mod p' \end{cases}$$

24 $\forall c_p, c_q, c_S$

$$\begin{cases} \tilde{S} \equiv \tilde{S}'^{c_p c_q c_S}_q \equiv 1^{c_p c_q c_S} \equiv 1 \mod q' \\ \tilde{S} \equiv \tilde{S}'^{c_p c_q c_S}_p \not\equiv 1 \mod p' \end{cases}$$

$\Rightarrow$ $\gcd(\tilde{S} - 1, N) = q$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 1

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Randomizing fault:

- $M = 1$

15 $\tilde{S}'_p \leftarrow \epsilon \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q \equiv 1 \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p \not\equiv 1 \mod p' \end{cases}$$

24 $\forall c_p, c_q, c_S$

$$\begin{cases} \tilde{S} \equiv \tilde{S}'^{c_p c_q c_S}_q \equiv 1^{c_p c_q c_S} \equiv 1 \mod q' \\ \tilde{S} \equiv \tilde{S}'^{c_p c_q c_S}_p \not\equiv 1 \mod p' \end{cases}$$

$\Rightarrow \gcd(\tilde{S} - 1, N) = q$

Available fault targets:

- Steps 5, 8, 11, 14, 15 or 18

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

Choose a small random integer r.
$N = p \cdot q$
$p' = p \cdot r^2$
$i_{pr} = p^{-1} \mod r^2$
$M_p = M \mod p'$
$B_p = p \cdot i_{pr}$
$A_p = 1 - B_p \mod p'$
$M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
$q' = q \cdot r^2$
$i_{qr} = q^{-1} \mod r^2$
$M_q = M \mod q'$
$B_q = q \cdot i_{qr}$
$A_q = 1 - B_q \mod q'$
$M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
$S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
$S_{pr} = 1 + d_p \cdot r$
$c_p = M'_p + N - M + 1 \mod p$
$S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
$S_{qr} = 1 + d_q \cdot r$
$c_q = M'_q + N - M + 1 \mod q$
$S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
$S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
$c_S = S - S_r + 1 \mod r^2$
return $S = S'^{c_p c_q c_S} \mod N$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 2

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Stuck-at-0 fault:

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 2

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Stuck-at-0 fault:

15 $\tilde{S}'_p \leftarrow 0 \mod p'$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 2

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Stuck-at-0 fault:

15 $\tilde{S}'_p \leftarrow 0 \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q = {m_q}^{dq} \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p = 0 \mod p' \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 2

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Stuck-at-0 fault:

15 $\tilde{S}'_p \leftarrow 0 \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q = {m_q}^{dq} \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p = 0 \mod p' \end{cases}$$

24 $\forall c_p, c_q, c_S$

$$\begin{cases} \tilde{S} \equiv {\tilde{S}'_q}^{c_p c_q c_S} \not\equiv 0 \mod q' \\ \tilde{S} \equiv {\tilde{S}'_p}^{c_p c_q c_S} \equiv 0 \mod p' \end{cases}$$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 2

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Stuck-at-0 fault:

15 $\tilde{S}'_p \leftarrow 0 \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q = {m_q}^{dq} \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p = 0 \mod p' \end{cases}$$

24 $\forall c_p, c_q, c_S$

$$\begin{cases} \tilde{S} \equiv {\tilde{S}'_q}^{c_p c_q c_S} \not\equiv 0 \mod q' \\ \tilde{S} \equiv {\tilde{S}'_p}^{c_p c_q c_S} \equiv 0 \mod p' \end{cases}$$

$\Rightarrow$ $\gcd(\tilde{S}, N) = p$

oberthur TECHNOLOGIES THE M COMPANY

Translated Vigilant: Attack 2

Algorithm 11: CRT-RSA with our simplified Vigilant's countermeasure, under its infective avatar

Input : Message M, key (p, q, d_p, d_q, i_q)
Output: Signature $M^d \mod N$, or a random value in $\mathbb{Z}_N$

1 Choose a small random integer r.
2 $N = p \cdot q$
3 $p' = p \cdot r^2$
4 $i_{pr} = p^{-1} \mod r^2$
5 $M_p = M \mod p'$
6 $B_p = p \cdot i_{pr}$
7 $A_p = 1 - B_p \mod p'$
8 $M'_p = A_p \cdot M_p + B_p \cdot (1 + r) \mod p'$
9 $q' = q \cdot r^2$
10 $i_{qr} = q^{-1} \mod r^2$
11 $M_q = M \mod q'$
12 $B_q = q \cdot i_{qr}$
13 $A_q = 1 - B_q \mod q'$
14 $M'_q = A_q \cdot M_q + B_q \cdot (1 + r) \mod q'$
15 $S'_p = {M'_p}^{d_p \mod \varphi(p')} \mod p'$
16 $S_{pr} = 1 + d_p \cdot r$
17 $c_p = M'_p + N - M + 1 \mod p$
18 $S'_q = {M'_q}^{d_q \mod \varphi(q')} \mod q'$
19 $S_{qr} = 1 + d_q \cdot r$
20 $c_q = M'_q + N - M + 1 \mod q$
21 $S' = S'_q + q \cdot (i_q \cdot (S'_p - S'_q) \mod p')$
22 $S_r = S_{qr} + q \cdot (i_q \cdot (S_{pr} - S_{qr}) \mod p')$
23 $c_S = S - S_r + 1 \mod r^2$
24 **return** $S = S'^{c_p c_q c_S} \mod N$

Stuck-at-0 fault:

15 $\tilde{S}'_p \leftarrow 0 \mod p'$

21

$$\begin{cases} \tilde{S}' \equiv S'_q = {m_q}^{dq} \mod q' \\ \tilde{S}' \equiv \tilde{S}'_p = 0 \mod p' \end{cases}$$

24 $\forall c_p, c_q, c_S$

$$\begin{cases} \tilde{S} \equiv {\tilde{S}'_q}^{c_p c_q c_S} \not\equiv 0 \mod q' \\ \tilde{S} \equiv {\tilde{S}'_p}^{c_p c_q c_S} \equiv 0 \mod p' \end{cases}$$

$\Rightarrow$ $\gcd(\tilde{S}, N) = p$

The fault can be injected into:

- Step 5, 8, 11, 14, 15 or 18

oberthur TECHNOLOGIES THE M COMPANY

Overview

oberthur TECHNOLOGIES THE M COMPANY

Conclusion

Warning The infective countermeasures of FDTC 2014 insecure.

Warning The security proof of translation method insecure.

- The translation does not solve the problem of security of an infective countermeasure.

Thanks