Modern Cryptology: Standards Are Not Enough

Ing. Tomáš Rosa

Doctoral Thesis Presentation

Department of Computer Science and Engineering
Faculty of Electrical Engineering,
Czech Technical University in Prague

Agenda

- Introduction
  - Side Channels – Basic Definitions
- Results Presentation
  - Overview
  - Key Ideas
  - Theoretical & Practical Merit
- Thesis Summary

What has been done

- Cryptographic security of various industry standards was investigated.
- The rapidly growing theory of side channels was successfully deployed.
- Certain new viewpoints of security requirements were introduced.

Side Channel

- Any undesirable way of information exchange between a cryptographic module and its neighbourhood.
  - Timing
  - Power
  - Electromagnetic
  - Fault
  - Kleptographic

Side channel

Side Channel Analysis

- A procedure of getting information from a side channel.
  - Simple
  - Differential

  } Analysis

Side Information

- The information obtained by a side channel analysis.
  - Particular key bits.
  - Condition status.
  - Hamming weights of operands.
  - A result of a faulty computation.

Side Channel Attack

- A process of using side information to attack a cryptographic module.
  - Timing
  - Power
  - Electromagnetic
  - Fault
  - Kleptographic

} Attack

Attack on OpenPGP Key Storage

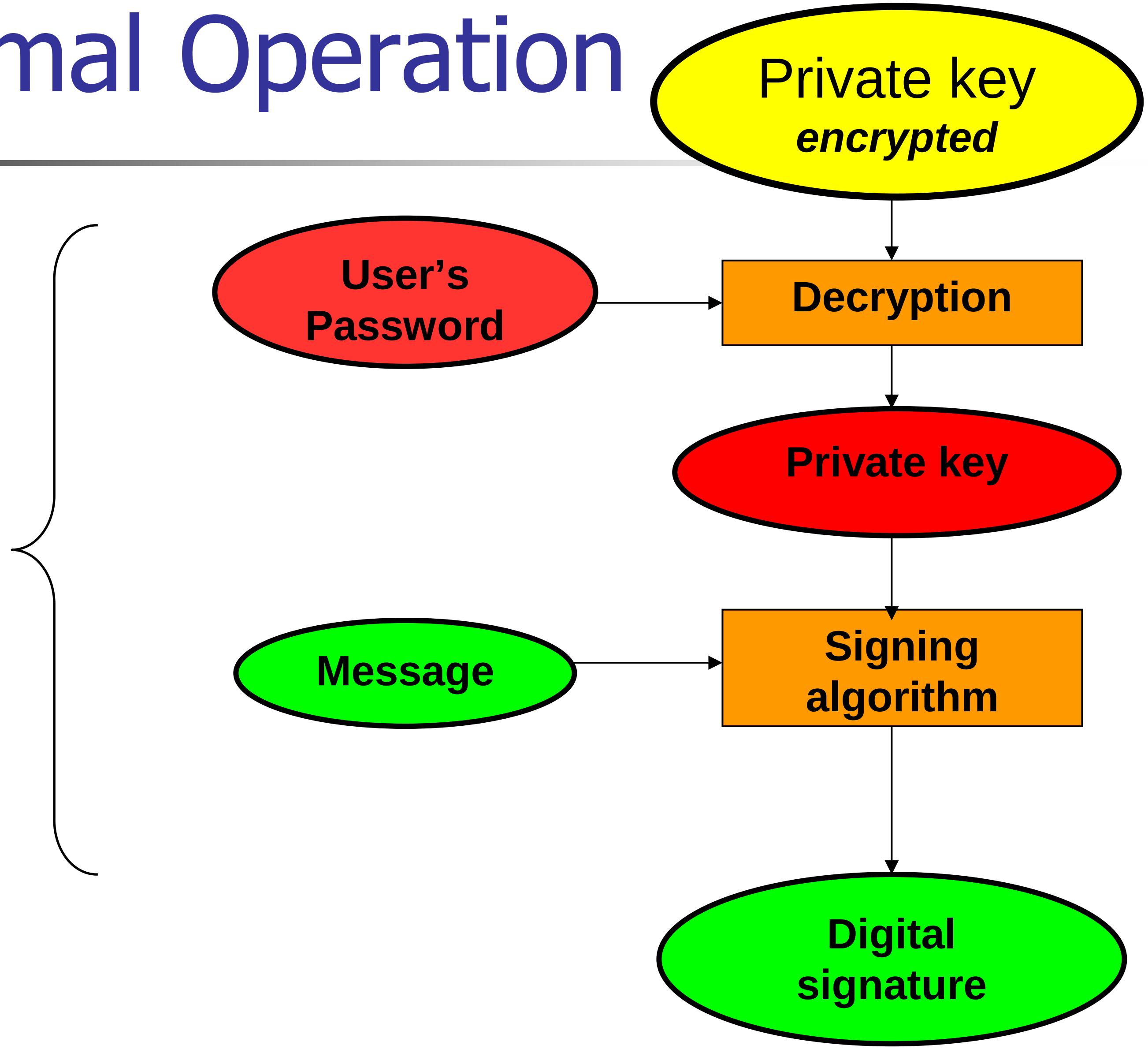
Thesis – Part B

Overview

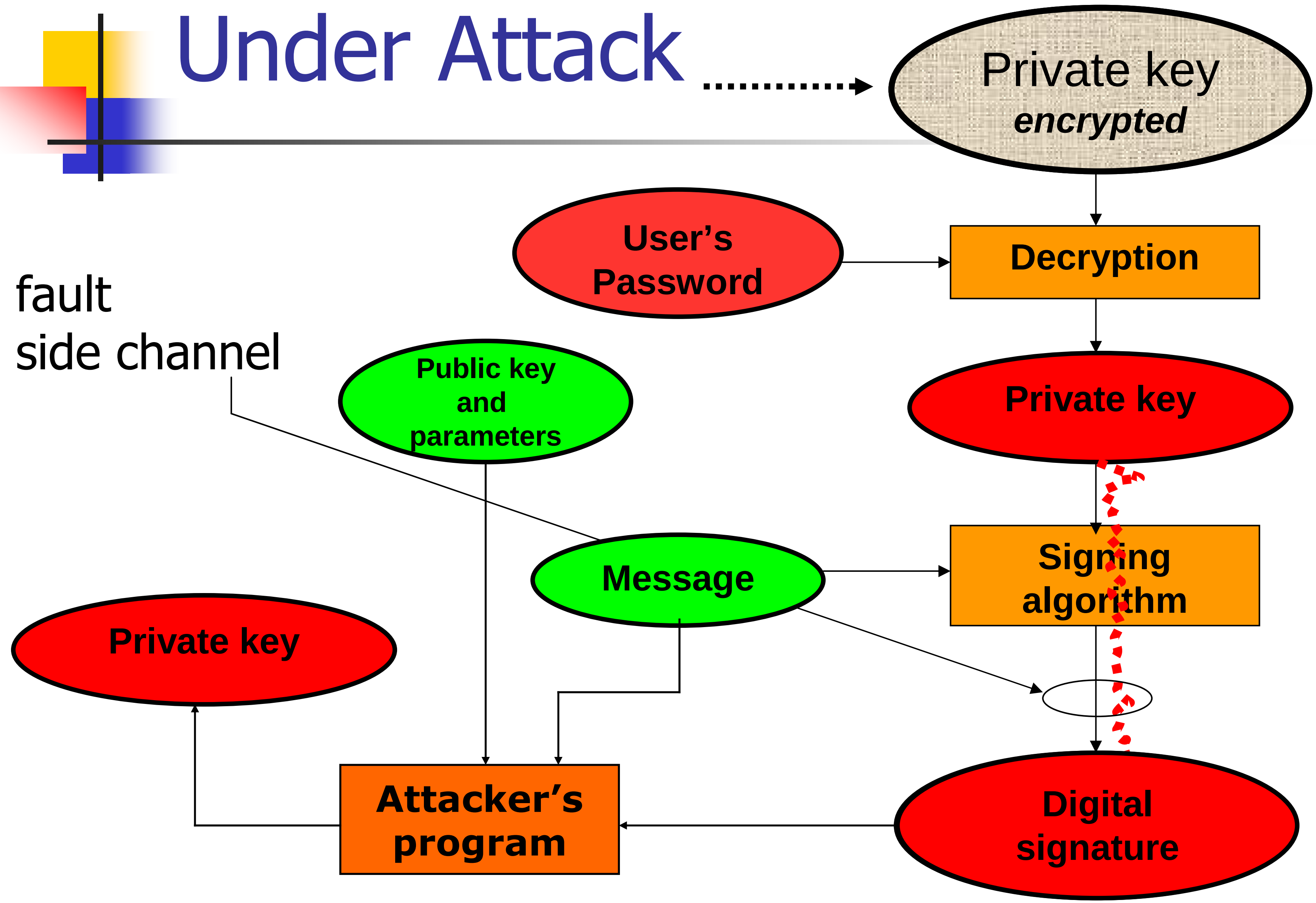
- Insufficient/missing integrity checks of encrypted private keys were found in the OpenPGP standard (RFC 2440).
- Modification of a key record induces a leakage of the complete private key.
  - The attack concerns not only the keys stored locally in a workstation. It affects the keys being transferred via *net, as well.
- This is a special kind of fault attack.

Normal Operation

Signing a message



Under Attack



Attack on RSA Keys

- Extends Lenstra's original fault attack.
  - A usable faulty computation can also be induced by corrupting the private key values before the computation starts.
- OpenPGP stores the key as (p, q, $pInv$, d).
  - There is an improper integrity check of $pInv$. By affecting a ciphertext image of $pInv$, the attacker can change it, so that $p * pInv \bmod q \neq 1$ with a high probability.
  - Such a modification allows computation of the whole key from only one faulty signature made.
- For the faulty signature s' we have: $[(s')^e - m] \bmod p = 0$, while $[(s')^e - m] \bmod q \neq 0$.
  - From here, $p = \gcd(N, (s')^e - m)$, where N, $N = pq$, is the public RSA modulus.

Attack on DSA Keys

- Private key record contains (among others):
  - Encrypted values:
    - private key x, $0 < x < q$
  - Unencrypted values without any cryptographic integrity check:
    - public parameters (p, q, g)
    - public key y, $y = g^x \bmod p$

- For a signature (r, s) it holds that:
  - $r = (g^k \bmod p) \bmod q$, $k \in_R \{1, ..., q - 1\}$
  - $s = (h(m) + xr)k^{-1} \bmod q$, $h =^{def}$ SHA-1

- For every DSA instance, there is a modification of the values (p, q, g) to (p', q, g'), such that the private key x can be easily computed from only one faulty signature (r', s').
  - Main idea: $2^{158} < p' < q$, g' generates $\boldsymbol{Z}_{p'}^*$, $(p' - 1)$ is smooth.

Theoretical Merit

- Integrity preservation is an important factor for preserving privacy.
  - These two factors were usually regarded separately.
  - Fault attacks on RSA-CRT can be induced by a private key modification.
- All values that are processed together with secret keys (including parts of that key) must satisfy appropriate integrity constraints.

Practical Merit

- Influence on OpenPGP-based programs.
  - PGP 8.0.2 was updated to prevent the attack.
  - GnuPG was designed having the attack on mind.
- Inspired an analysis of certain parts of PKCS#11.
  - Presented by J. Clulow at CHES 2003.
- Influenced a development of cryptographic devices for the Czech NSA.

Side Channel Attacks on Certain RSA Schemes

Thesis – Part C

Overview

- Side-channel attack on an “OAEP-shielded” part of the RSAES-OAEP scheme.
  - The scheme is regarded as a safer ancestor of a weaker method called RSAES-PKCS1-v1_5.
- Furthermore, we point out several design flaws in the RSA-KEM scheme.
  - RSA-KEM is a candidate for an ISO standard for public key encryption.
  - We show a misconception in private key handling and emphasize its inner vulnerability to fault side channel attacks.

Place of Our Attack on RSAES-OAEP

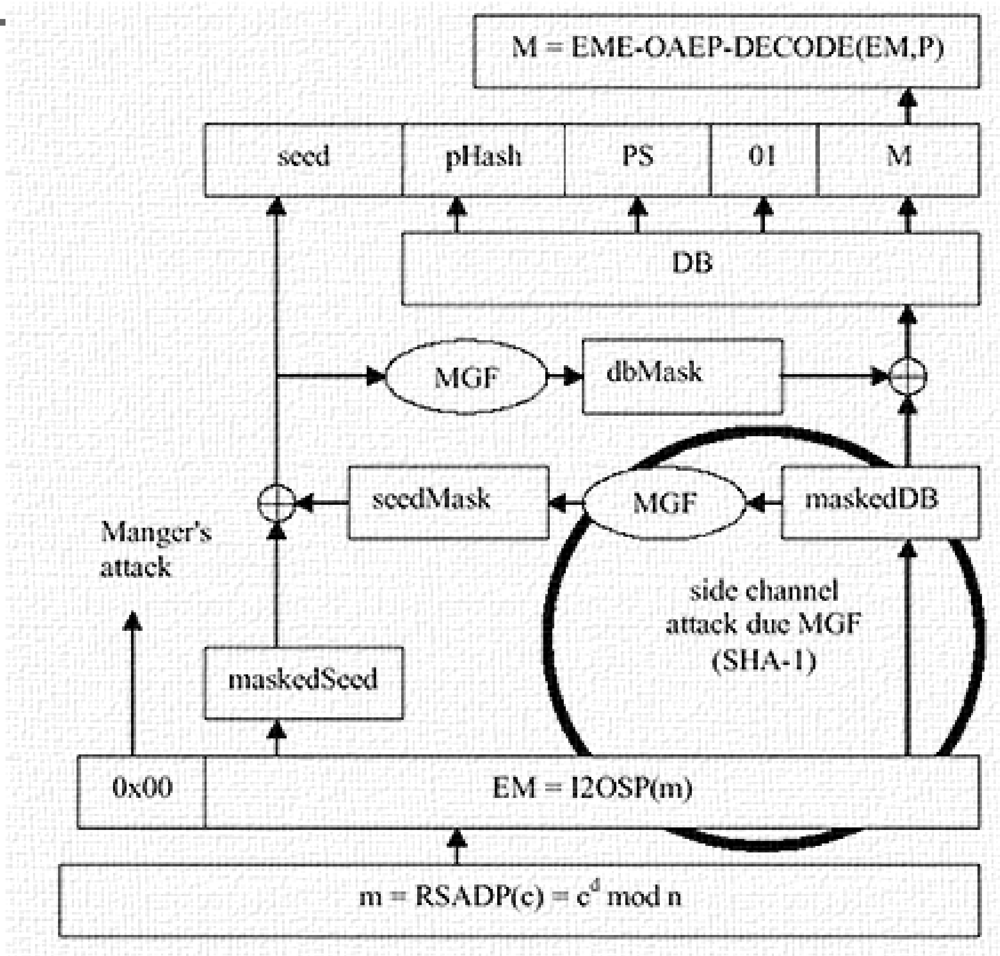


Fig. 1. New side channel attack against RSAES-OAEP

Hamming Weight Leakage

- We may reasonably assume that Hamming weights of arguments of operations in the RSAES-OAEP scheme can leak out through a power side channel.

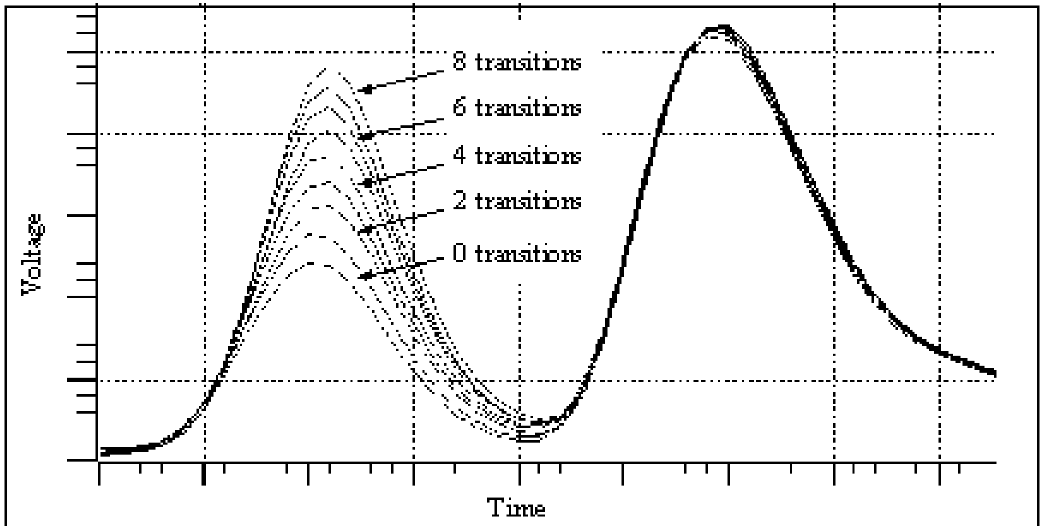


FIGURE 2. **Number of Bit Transitions versus Power Consumption**
These results show how the data effects the power levels. The nine overlayed waveforms correspond to the power traces of different data being accessed by an LDA instruction. These results were obtained by averaging the power signals across 500 samples in order to reduce the noise content. The difference in voltage between i transitions and i+1 transitions is about 6.5 mV.

([17])

Exploiting the Leakage for an Attack

- First, we build an lsb-oracle for getting lsb(m).
  - We prepare two special ciphertexts c and c'.
  - If lsb(m) = 0 then certain Hamming weights observed for c and c' are related linearly.
  - If lsb(m) = 1 then the probability of a random linear relationship is very low.
    - **From here we get the oracle $O_{lsb}(c)$ for lsb(m).**
- Second, we use the lsb-oracle O_{lsb} and deploy general purpose RSA-inversion algorithm.
  - It takes $\boldsymbol{O}(\text{Log}_2(N)^2 adv^{-2})$ oracle calls, where N is the RSA modulus and adv is the advantage describing O_{lsb} accuracy ([11]).
    - $adv = |P[\text{lsb}(m) = O_{lsb}(c)] - ½|$

Hamming Weights Relations for lsb(m) = 0

Table 1. Possible relations among random variables W and W' when $W_{10,8} = 0$

$W_{9,0}$	$W_{8,0}$	$W_{7,0}$	Possible relations		
0	0	0	$w(W_{10}') = w(W_{10})$	$w(W_9') = w(W_9)$	$w(W_8') = w(W_8)$
0	0	1	$w(W_{10}') = w(W_{10})$	$w(W_9') = w(W_9)$	$w(W_8') = w(W_8) +1$
0	1	0	$w(W_{10}') = w(W_{10})$	$w(W_9') = w(W_9) +1$	$w(W_8') = w(W_8) -1$
0	1	1	$w(W_{10}') = w(W_{10})$	$w(W_9') = w(W_9) +1$	$w(W_8') = w(W_8)$
1	0	0	$w(W_{10}') = w(W_{10}) +1$	$w(W_9') = w(W_9) -1$	$w(W_8') = w(W_8)$
1	0	1	$w(W_{10}') = w(W_{10}) +1$	$w(W_9') = w(W_9) -1$	$w(W_8') = w(W_8) +1$
1	1	0	$w(W_{10}') = w(W_{10}) +1$	$w(W_9') = w(W_9)$	$w(W_8') = w(W_8) -1$
1	1	1	$w(W_{10}') = w(W_{10}) +1$	$w(W_9') = w(W_9)$	$w(W_8') = w(W_8)$

The three types: ↑   -   ↓

RSA Confirmation Oracle (RSA-CO)

- An RSA-CO confirms whether integers r, y satisfy $r = y^d \bmod N$.
  - Here, (d, N) are the values *regarded* by the module as the private key.
  - It can be generalized for any encryption scheme (the condition tested may also be more general).
- If there are faults then the RSA-CO reveals nontrivial information about the private key.

Building RSA-CO on RSA-KEM Properties

- We use the properties of the whole hybrid scheme H-PKE.
  - There is no integrity check for the RSA plaintext (*r*).
    - Obviously, this is a good property against CCA_2, however it also implies that any resulting RSA plaintext will be used for a symmetrical decryption.
    - Integrity controls applied on the message decrypted symmetrically then confirms our guess of *r*.
- Summary: What makes the RSA-KEM stronger in other areas, that makes it vulnerable to fault attacks, on the other hand.

Using RSA-CO for Attacks on RSA-KEM

- The modulus *N* is not regarded as an integral part of the private key (*d*, *N*).
  - Therefore, changing (*d*, *N*) for (*d*, *N′*) can be possible.
  - Such a change together with an RSA-CO leads to the complete private key disclosure.
- Furthermore, an RSA-CO can be used for porting other known fault attacks on RSA.
  - Exploiting bit faults in the private exponent *d*, for instance.

Theoretical Merit

- Hamming weight leakage can be used for an RSAES-OAEP inversion.
  - First public side-channel attack on an "OAEP-shielded" part of RSAES-OAEP scheme.
- The notion of Confirmation Oracle was introduced for RSA.
  - Certain parts of an ISO candidate RSA-KEM were shown to be vulnerable.
- Padding methods themselves cannot fully defeat side channel attacks.

Practical Merit

- Side channel leakage must also be investigated for an "OAEP-shielded" part of the RSA-OAEP scheme.
- The RSA-KEM scheme shall be updated.
- Inspired an analysis of certain parts of PKCS#11.
  - Presented by J. Clulow at CHES 2003.
- The work has been appreciated in the smart card industry.

Strengthened CBC Mode

Thesis – Part D

Overview

- Vaudenay showed that a CBC encryption mode with a PKCS#5 padding is vulnerable through fault side channel attack.
  - His countermeasures, however, don’t fit into the semantics of contemporary cryptographic APIs.
- We propose several modifications of CBC mode with respect to the final block encryption.
  - They do fit into the semantics of cryptographic APIs.
  - They objective is to de-linearize and randomly mask the influence of the penultimate cipherblock on the final block encryption.

Where Was the Vulnerability

- Main Issue of CBC-PKCS#5
  - There is a Confirmation Oracle telling us for arbitrary chosen y, γ and given key K if:
    - $x \in$ ***PAD*** for $x = D_K(y) \oplus \gamma$,
    - ***PAD*** = {*||01, *||0202, *||030303, ...}
      - The length of every x, $x \in$ ***PAD***, equals to the block length of the particular CBC mode.
  - Such a CO can be used to compute $D_K(y)$ effectively.
    - First, we search for γ_1 inducing $x \in$ {*||01}, then for γ_2 inducing $x \in$ {*||0202}, etc.

Our Approach

- Randomize the influence of c_{N-1} on m_N.
  - Confirmation oracle is no longer useful.
- We do that by changing the encryption rules for the final CBC block m_N.
  - It preserves the whole semantics of the CBC mode, i.e.:
    - during the last block encryption, 1 or 2 blocks are returned,
    - during the last block decryption, 0 to B bytes is returned, where B is the block length.

Theoretical Merit

- The notion of Confirmation Oracle can usefully be adopted in symmetrical ciphers, as well.

- We proposed a “3rd kind” of countermeasure against Vaudenay’s attack:
  - 1. was using strict EtA concept – Encrypt-then-Authenticate
  - 2. was using special, error-free padding (c.f. Part E)
  - 3. is eliminating certain properties of CBC mode (predictable propagation of changes of ciphertext blocks)

Practical Merit

- A general countermeasure is suggested such that:
  - It eliminates Vaudenay's attack.
  - It does not introduce new practical weaknesses.
  - It is fully compatible with contemporary cryptographic APIs.
- Deployed in projects for the Czech NSA.

Side Channel Attack on PKCS#7 with CBC Encryption

Thesis – Part E

Overview

- Attack on PKCS#7 messages equipped with such a padding scheme that was regarded as being resistant against Vaudenay’s attack on CBC-PKCS#5.
- Successfully exploits the notion of Confirmation Oracle.

Basic CBC Properties Recalled

- $P_{i+1} = D_K(C_{i+1}) \oplus C_i$, $i \geq 0$, $C_0 =^{\text{def}} IV$
  - Changes in cipherblock C_i propagate <u>linearly and deterministically</u> to changes of the plaintext block P_{i+1}.
  - No matter how strong the cipher is.
  - An effect of i^{th} block corruption vanishes starting by block $(i + 2)$.
    - It affects only P_i and P_{i+1}.

Exploiting the CBC Properties

- Plaintext formatting rules create fault side channels.
  - Checking these rules opens a door for Confirmation Oracles of various kinds.
  - These oracles are vital tools of modern cryptanalysis.
- According to PKCS#7.
  - We attacked messages of the type OCTET STRING.
  - The plaintext consists of: HEAD || DATA || PADDING.
  - HEAD contains (**TYPE**, **LENGTH**), TYPE $=^{def}$ 0x4, the length covers the <u>DATA field without PADDING</u>.
  - Checking the values in HEAD creates the Confirmation Oracle $PKCS\#7_{conf.}$
  - The oracle allows decryption of any captured message with a linear complexity $\boldsymbol{O}(n)$.

Theoretical Merit

- Security of the whole scheme (e.g. padding ∪ message format) must be evaluated.
  - The way of developing universally secure padding is somehow misleading. At least, it detracts an attention paid to the interaction of the CBC properties with the whole message format.
- EtA model shall be used with CBC whenever there are some formatting rules set for the plaintext.
  - EtA – Encrypt then Authenticate

Practical Merit

- Highly structured data formats encrypted by CBC may turn out vulnerable.
  - Example of format that shall be checked is S/MIME.
- Schemes using popular TLV formats encrypted with CBC shall be checked.
  - TLV – Type Length Value
    - Each record is labeled by its Type and Length. Its Value then follows.
- The observations written in the article led to an improvement of proprietary security modules for the banking sector.

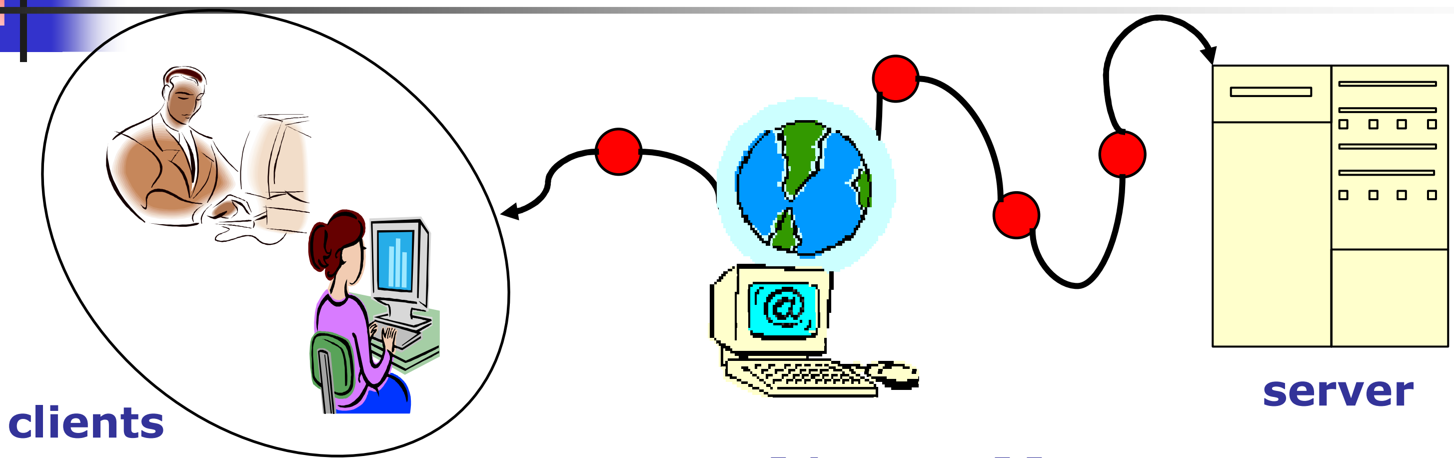
Attack on RSA in SSL/TLS

Thesis – Part F

Overview

- In 1998, Bleichenbacher shown an attack on RSAES-PKCS1-v1_5. SSL/TLS was regarded to be immune.
  - However, certain countermeasures were applied.
- We show an extension of Bleichenbacher's attack which applies to several SSL/TLS implementations and is practically feasible.
  - Therefore, SSL/TLS was not as immune as was deemed earlier.
- We also present several speed-ups of original Bleichenbacher's attack.

SSL/TLS
Session Setup



ClientHello

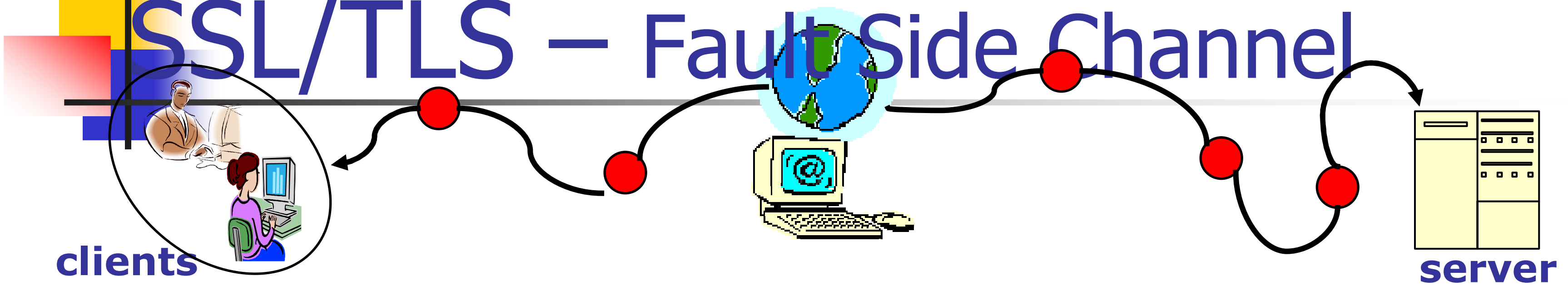
ServerHello, ..., ServerHelloDone

..., ClientKeyExchange, Finished

Finished

Transport channel

SSL/TLS – Fault Side Channel



ClientKeyExchange$_{RSA}$, Finished

C = [φ(*premaster-secret*)]^e mod N →

computation:

$P \leftarrow C^d \bmod N$

premaster-secret ← $\varphi^{-1}(P)$

if (exception in φ^{-1})

  premaster-secret ← RND(48)

else

  if(bad version of *premaster-secret*)

    "Alert-version"

Fault side channel

← **Finished/Alert**

Core of the Attack

- Seeing "Alert-version" we know that P = 00 02 ….
  - We write $P \in <E, F>$ for certain interval $<E, F>$.
- Let C_0 be the ciphertext we want to invert (with respect to RSA).
  - $C_0 = P_0^e \bmod N$
- Let $C = C_0 s^e \bmod N$, $s \in \mathbf{Z}$ and denote $P = C^d \bmod N$.
  - Note that P is still an unknown plaintext, $P = P_0 s \bmod N$.
- Now, seeing "Alert-version" we know that $E \leq sP_0 \bmod N \leq F$.
- From here, we get a useful information on P_0:
  - $(E+rN)/s \leq P_0 \leq (F+rN)/s$, for $r \in \mathbf{Z}$.
  - We obtain a set of intervals which may contain P_0.
- Using s producing "Alert-version", we can narrow the set of solutions for P_0 to get one particular value. This is then the inverse of C_0.
  - Each such s roughly halves the set of candidates for P_0.

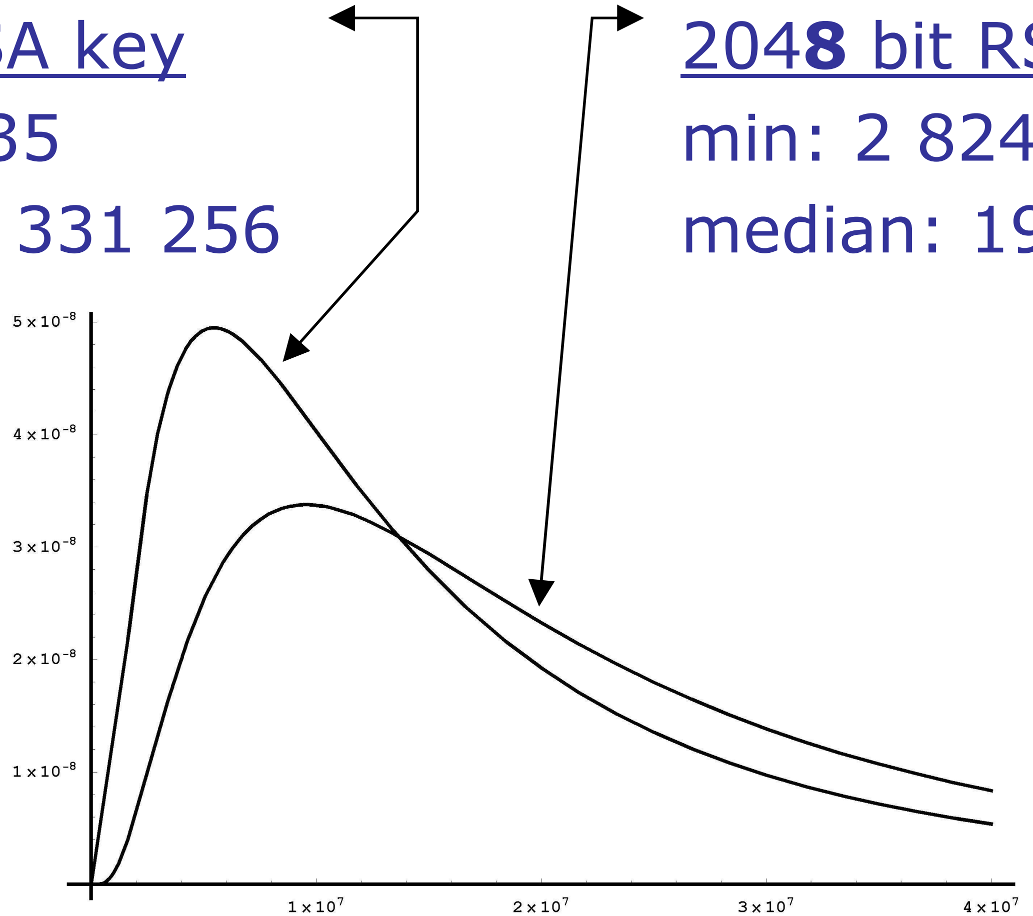
Note

- The version number check itself is a security measure.
  - However, its implementation created a vital fault side channel.
  - This channel allows an attacker to invert RSA transformation and decipher a private communication between a client and a server.
- Countermeasures based on indistinguishability between deciphered and random *premaster-secret*.
  - It is rather a subtle condition. Steps towards leaving PKCS1-v1_5 are desirable.

Amount of server calls

1024 bit RSA key
min: 815 835
median: 13 331 256

2048 bit RSA key
min: 2 824 986
median: 19 908 079



Theoretical Merit

- Bleichenbacher's attack can be extended on certain implementations of SSL/TLS.
- The attack is practically feasible in order of several days effort.
- Several countermeasures were proposed and discussed.

Practical Merit

- The discovery hit approx. 2/3 of world internet servers and it is echoed as one of the major reasons for upgrading server's software (worldwide).

Key-collisions in (EC)DSA

Thesis – Part F

Overview

- A (EC)DSA signature itself is not uniquely linked to a particular signatory.
- For a given signature, we can find another potential signatory who could make that signature.
  - We call it a *k*-collision (key-collision).
  - Under the condition of a public key variance, we can also find a message collision.

Non-repudiation versus *k*-collision

- *The non-repudiation property of a given action allows an independent third party to make sure that a particular event did (or did not) occur.*
- Possible disputation: Who signed that message?
  - Quick answer: Both of them.
  - Obstacle: What if only one of them could do that in a given time? How to decide who signed it then?

Countermeasure
How to Avoid *k*-collisions

- There is no proper *k*-collision searching algorithm that allows the public parameters of *k*-colliding instances to be chosen independently.
  - Provided the (EC)DSA scheme is not broken.
- The public parameters should be chosen by a third independent party.

Theoretical Merit

- A plain (EC)DSA signature cannot be regarded as a fingerprint of the message signed and-or a signatory identity.
  - However, there is a technically feasible countermeasure preventing *k*-collision attacks.
- The non-repudiation property can be threatened even if we use a signature scheme that does prevent signature forgery.

Practical Merit

- There was a real application potentially vulnerable to this attack.
  - The attack was reported to authors of the Slovak electronic signature law and notices.
- Proper attention has to be paid to designing non-repudiation service in information systems.
  - In this model, an attacker is often the person who usually plays the role of a victim.

Thesis Summary

- What environment shall the designed scheme be used in?
- What is the easiest problem an attacker has to solve to break the module in some way?
- Undoubtedly, standards are not enough to fully solve these problems.

Thank You