Get Your Hands Off My Laptop: Physical Side-Channel Key-Extraction Attacks on PCs

Daniel Genkin
Technion and Tel Aviv University

Itamar Pipman
Tel Aviv University

Eran Tromer
Tel Aviv University

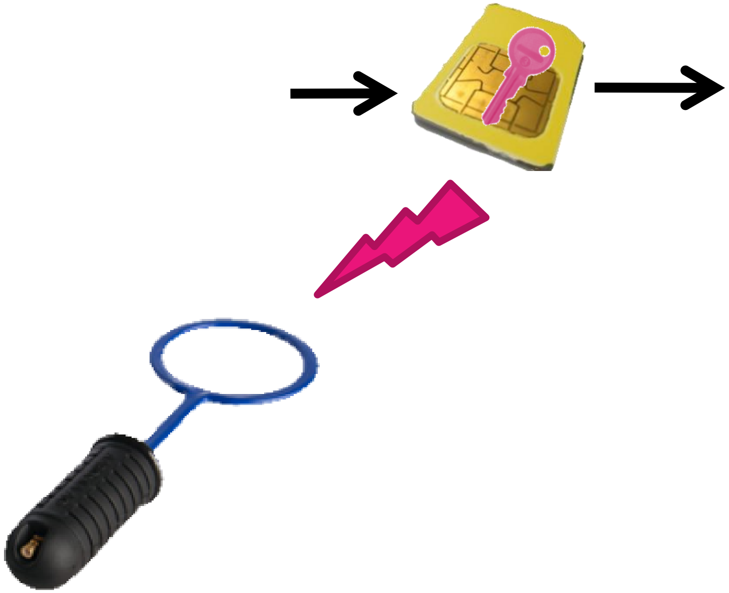
TEL AVIV UNIVERSITY

Laboratory for Experimental Information Security

Side channel attacks

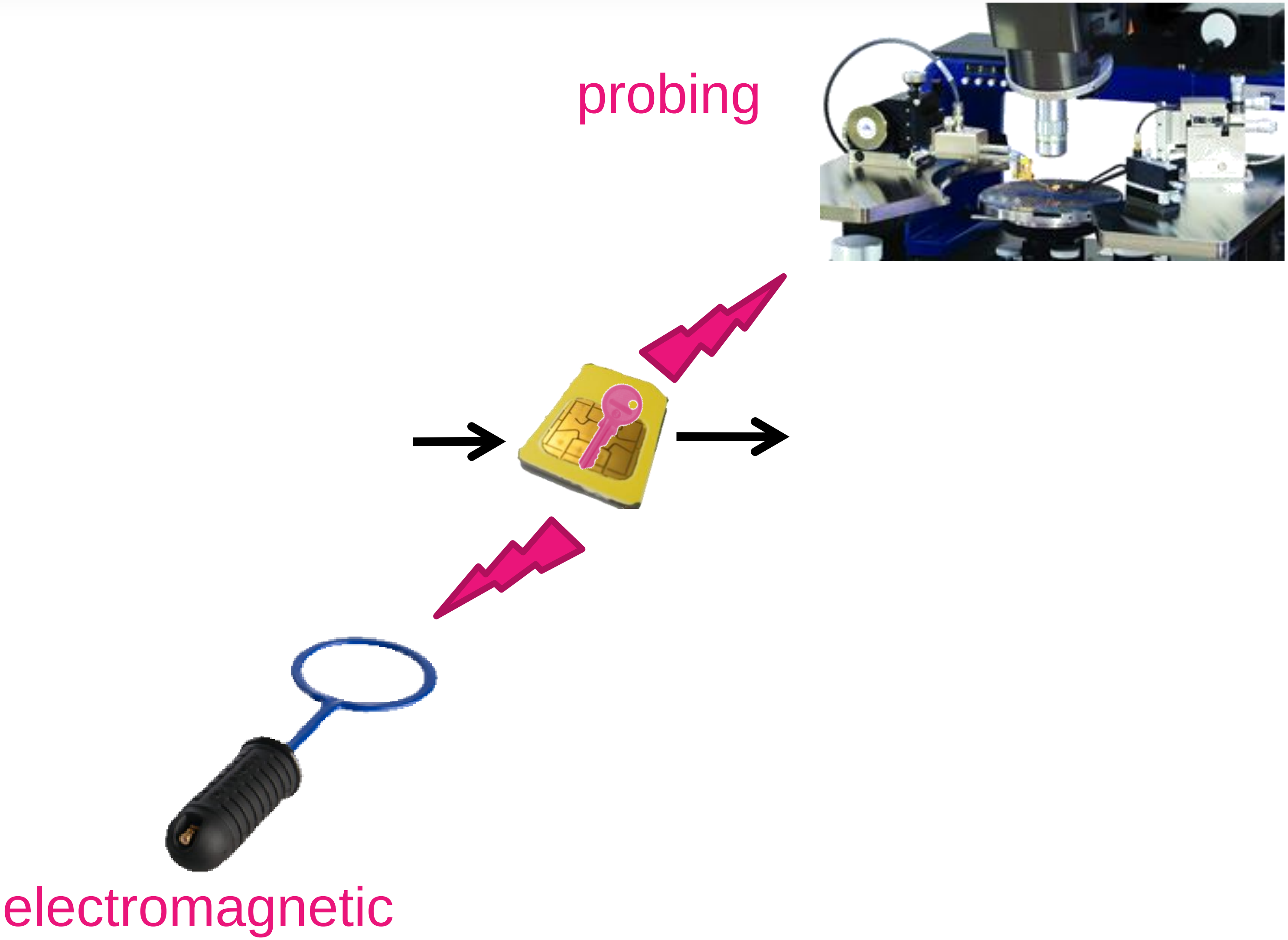
Side channel attacks

Side channel attacks



electromagnetic

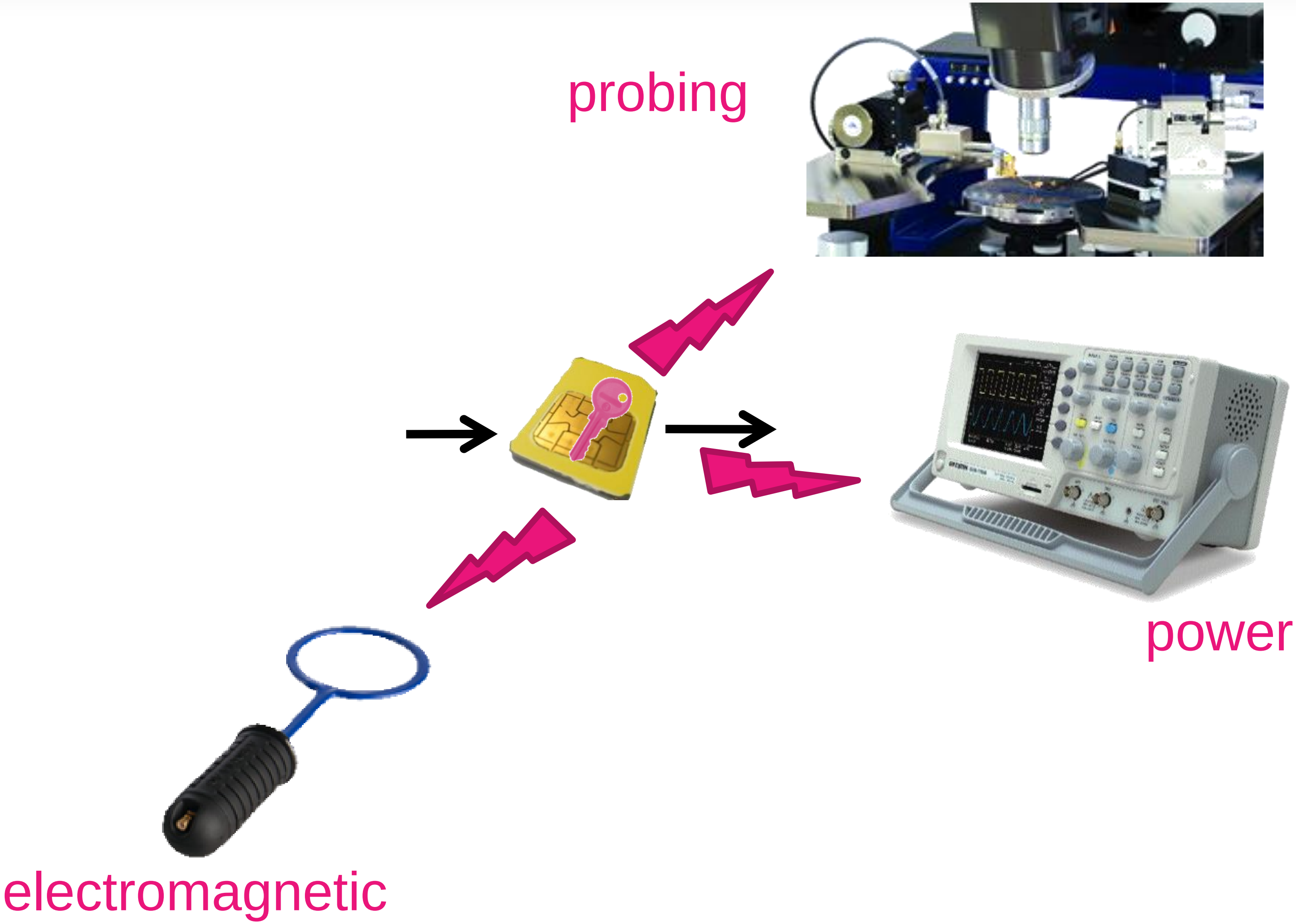
Side channel attacks



probing

electromagnetic

Side channel attacks

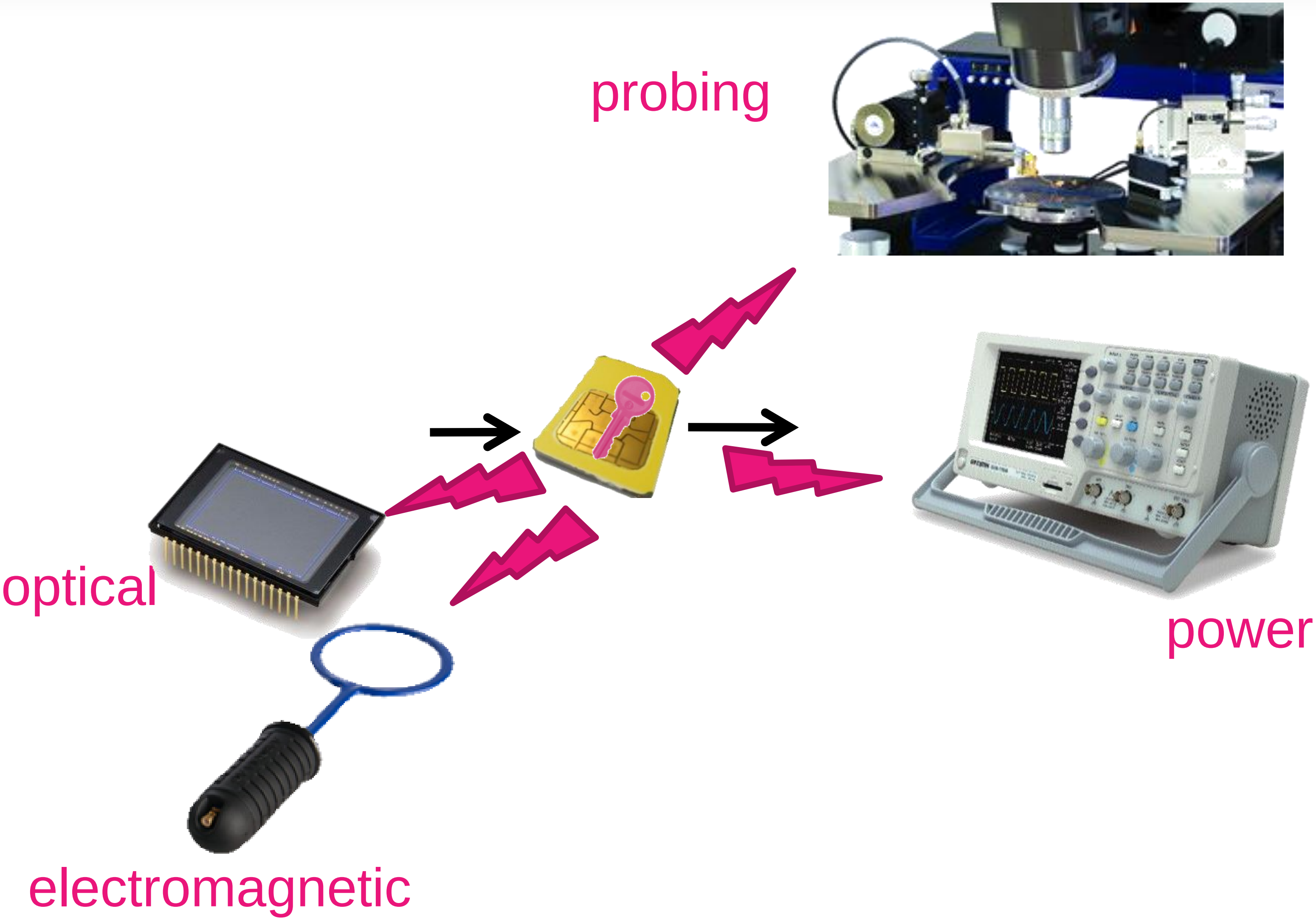


probing

power

electromagnetic

Side channel attacks



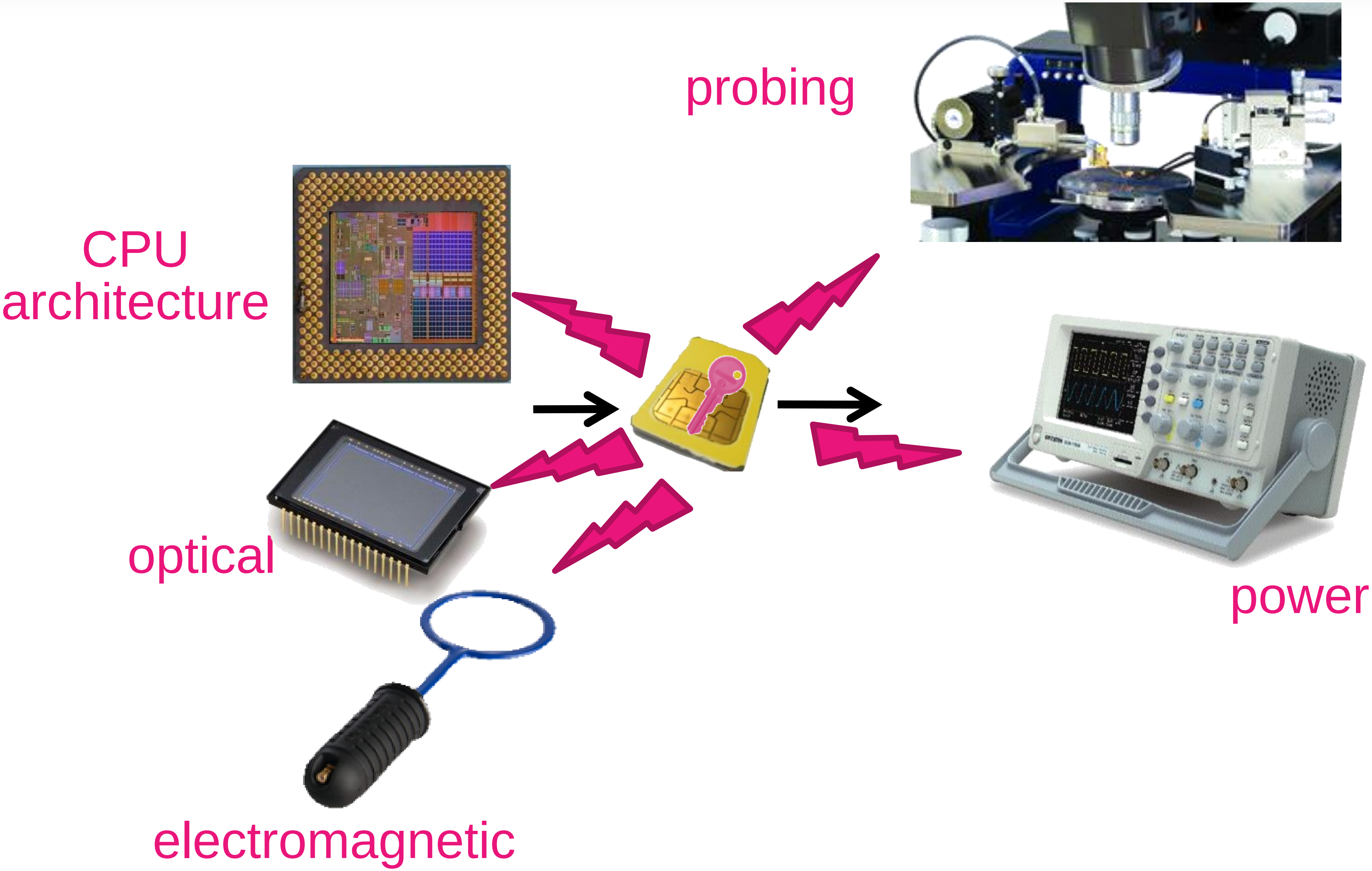
probing

optical

power

electromagnetic

Side channel attacks



probing

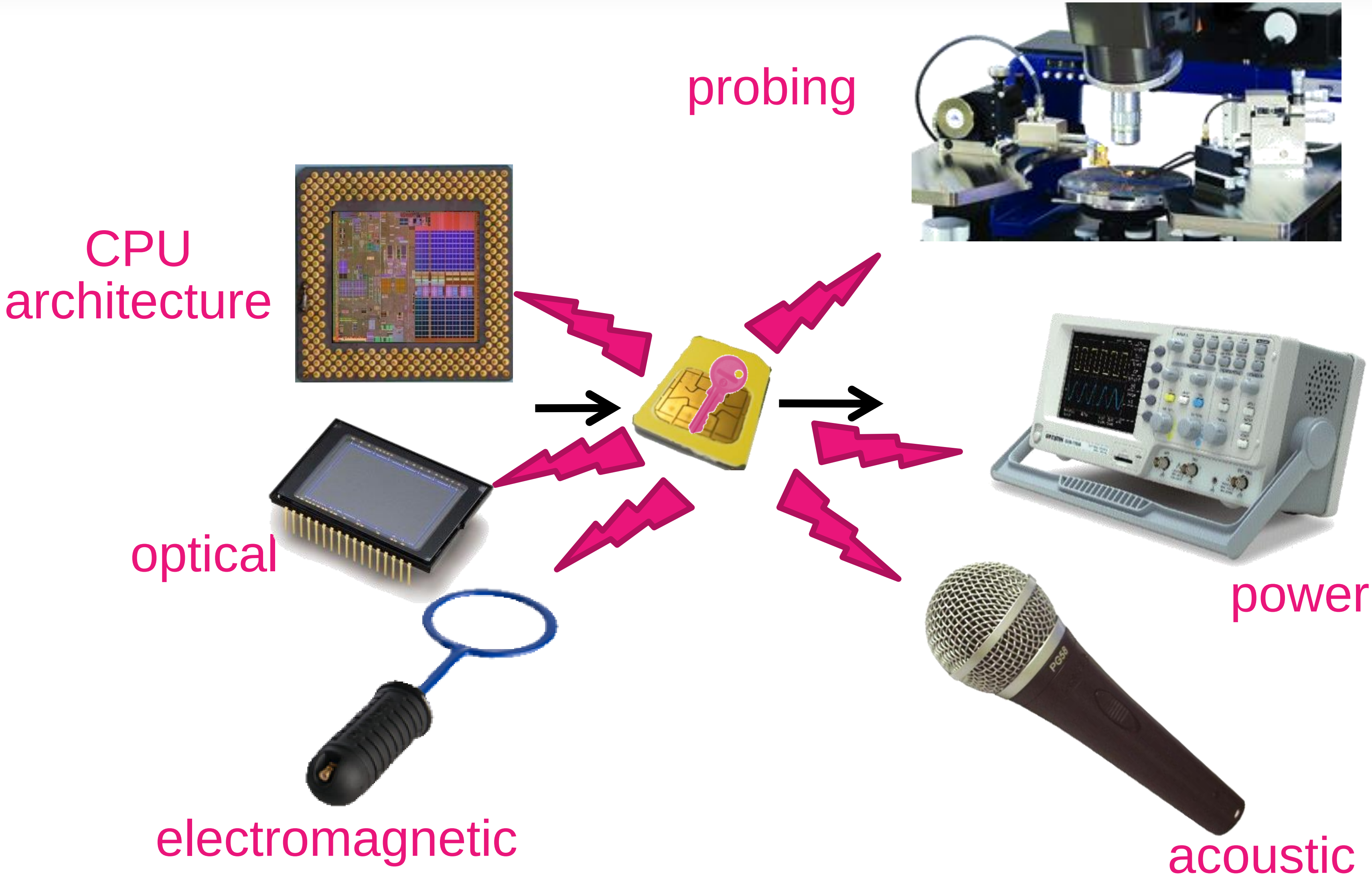
CPU architecture

optical

electromagnetic

power

Side channel attacks



probing

CPU architecture

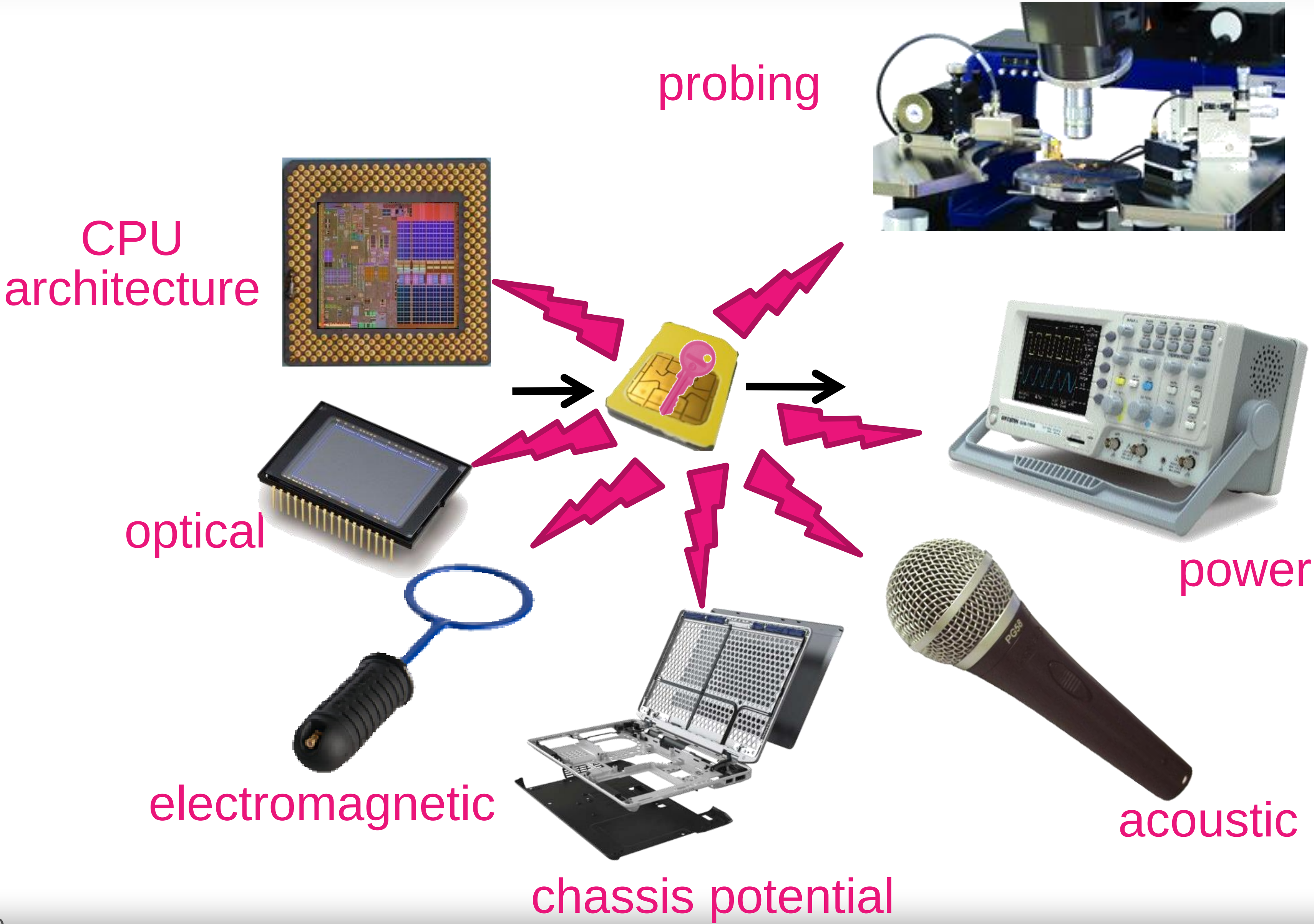
optical

electromagnetic

power

acoustic

Side channel attacks



probing

CPU architecture

optical

power

electromagnetic

acoustic

chassis potential

Traditional side channel attacks methodology

1. Grab/borrow/steal device

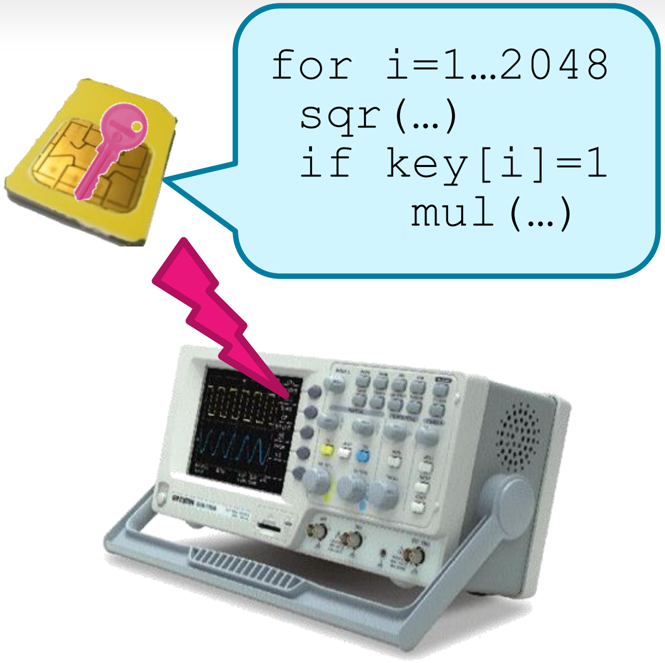
Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction

```
for i=1…2048
 sqr(…)
 if key[i]=1
     mul(…)
```

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)

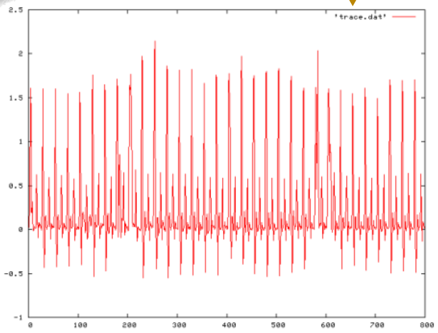


```
for i=1…2048
 sqr(…)
 if key[i]=1
     mul(…)
```

Traditional side channel attacks methodology

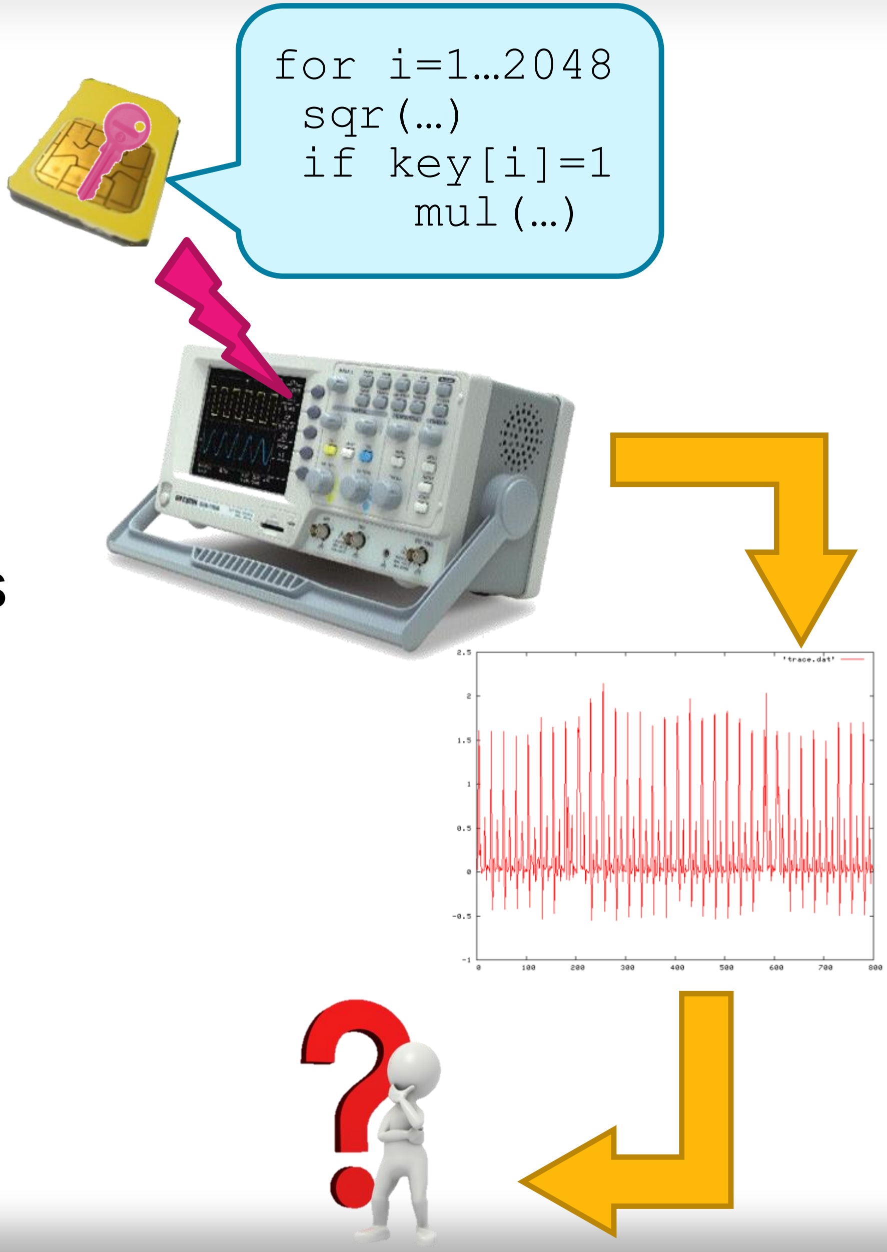
1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces

```
for i=1…2048
 sqr(…)
 if key[i]=1
     mul(…)
```



Traditional side channel attacks methodology

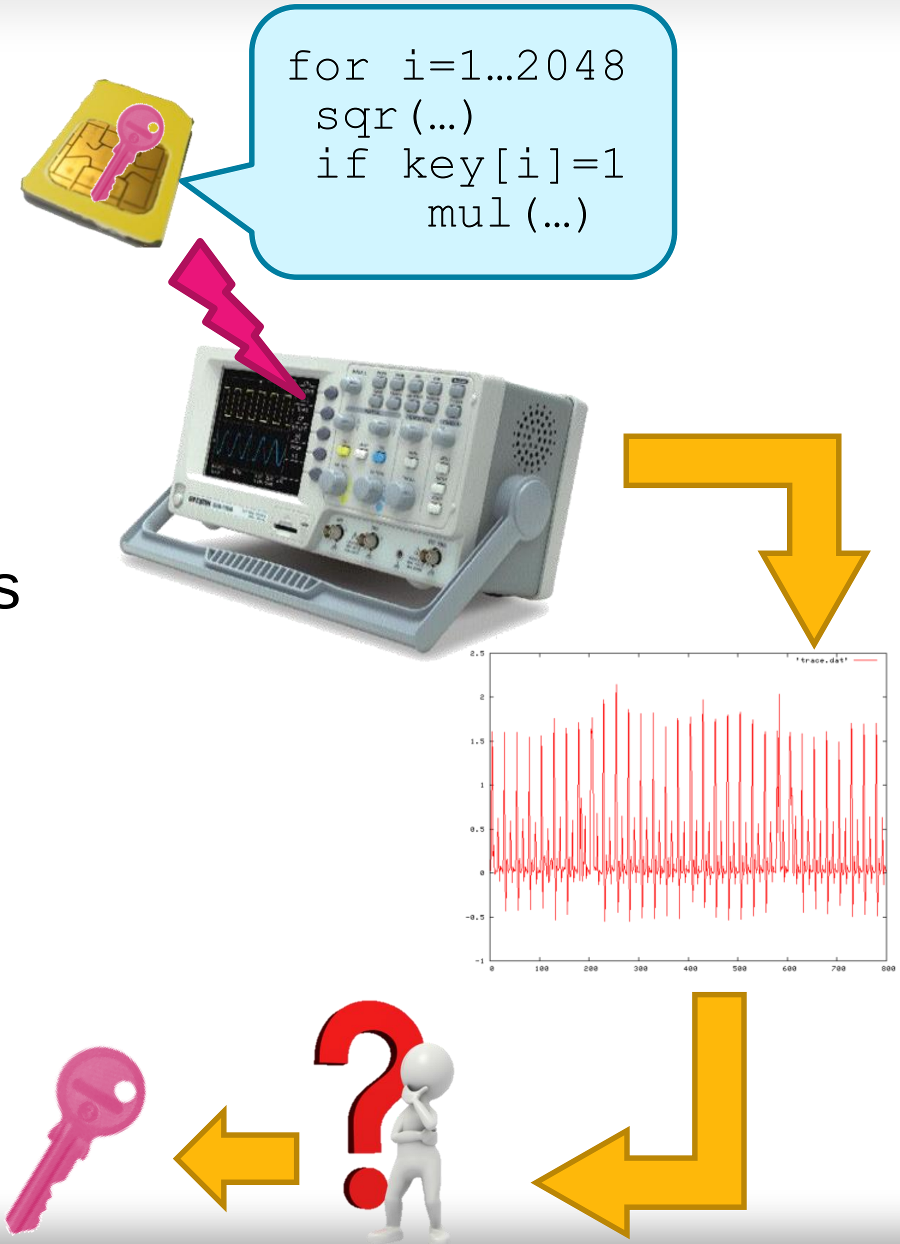
1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis



```
for i=1…2048
 sqr(…)
 if key[i]=1
    mul(…)
```

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis
6. Recover key

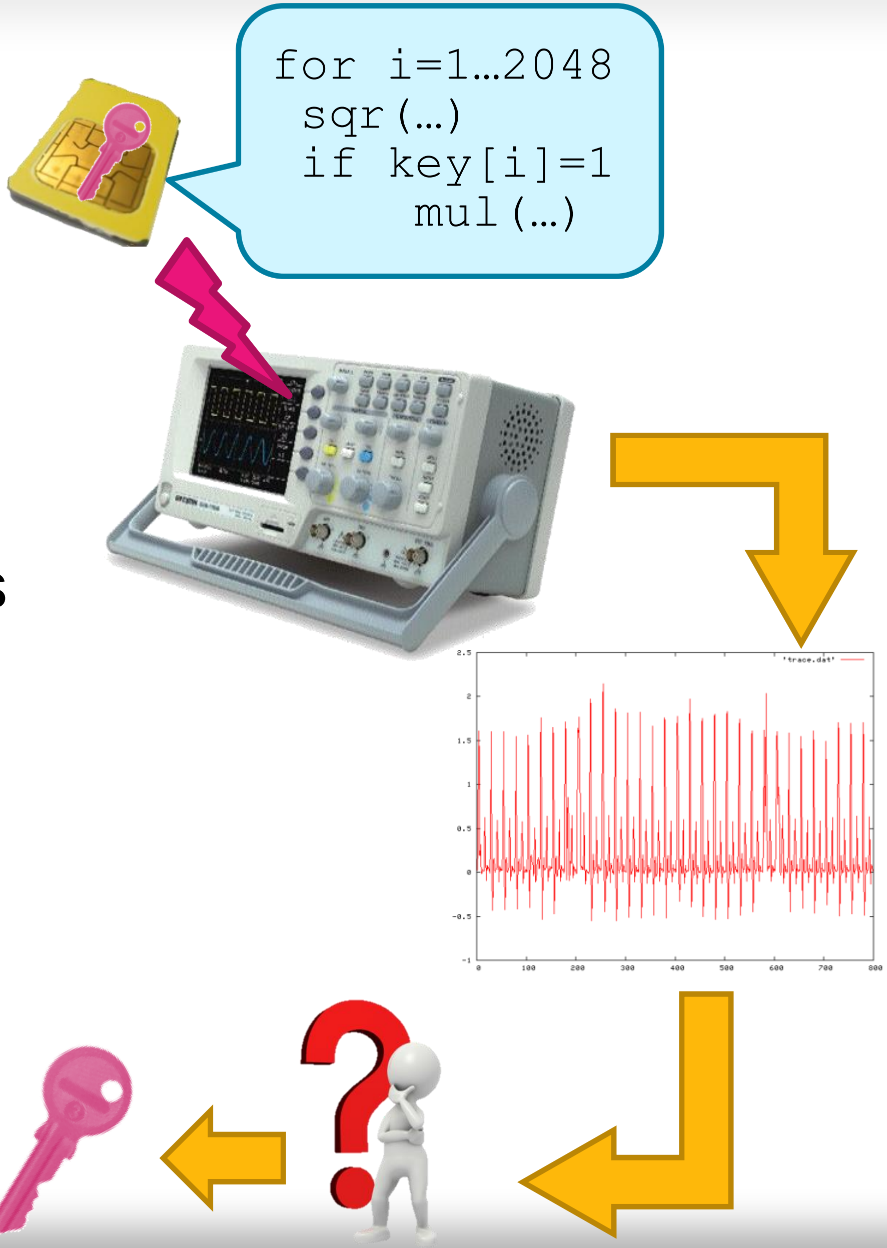


```
for i=1…2048
 sqr(…)
 if key[i]=1
     mul(…)
```

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis
6. Recover key

Hard for PCs



```
for i=1…2048
 sqr(…)
 if key[i]=1
     mul(…)
```

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis
6. Recover key

Hard for PCs ☹

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis
6. Recover key

Not handed out

Hard for PCs ☹

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis
6. Recover key

Hard for PCs ☹

Not handed out

vs.

Measuring a 2GHz PC requires expansive and bulky equipment (compared to a 100 MHz smart card)

vs.

1,000$

100,000$

Traditional side channel attacks methodology

1. Grab/borrow/steal device
2. Find key-dependent instruction
3. Record emanations using high-bandwidth equipment (> clock rate , PC: >2GHz)
4. Obtain traces
5. Signal and cryptanalytic analysis
6. Recover key

Not handed out

vs.

Measuring a 2GHz PC requires expansive and bulky equipment (compared to a 100 MHz smart card)

100,000$

vs.

1,000$

Complex electronics running complicated software (in parallel)

vs.

New channel: Chassis potential

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields?
  Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields?
  Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Computation

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis

Ground-potential analysis

- **Attenuating EMI emanations**
  "Unwanted currents or electromagnetic fields? Dump them to the circuit ground!"
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its "ground" potential fluctuates relative to the mains earth ground.

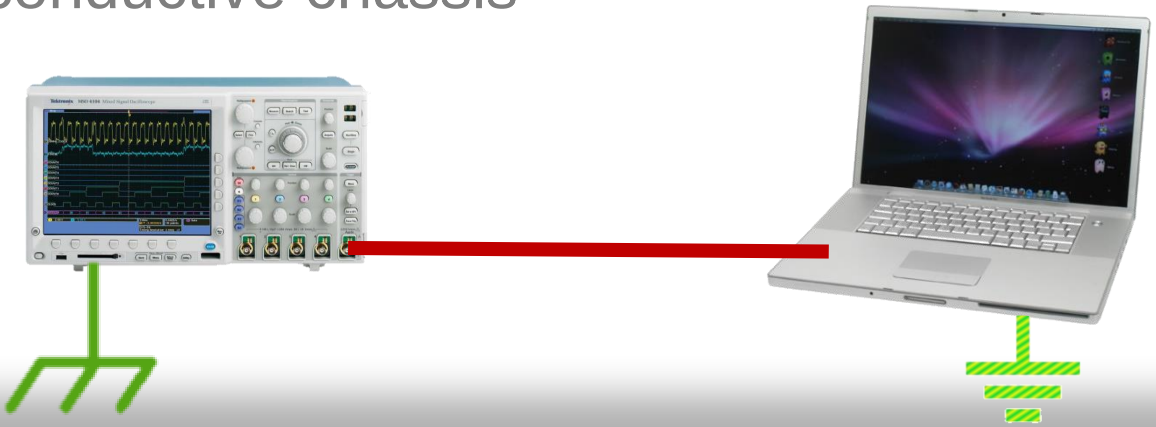
	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis

Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis



Ground-potential analysis

- **Attenuating EMI emanations**
  "Unwanted currents or electromagnetic fields? Dump them to the circuit ground!"
  (Bypass capacitors, RF shields, …)

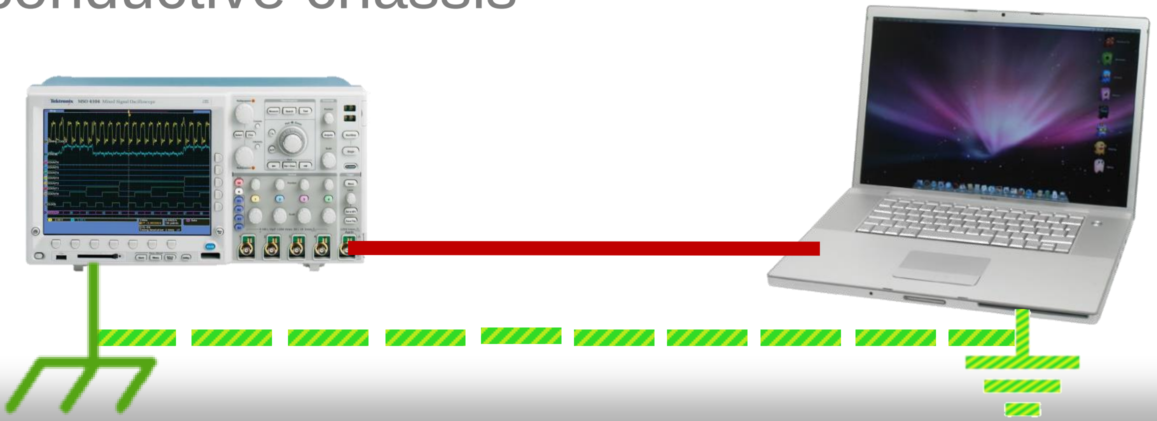
- Device is grounded, but its "ground" potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

connected to conductive chassis

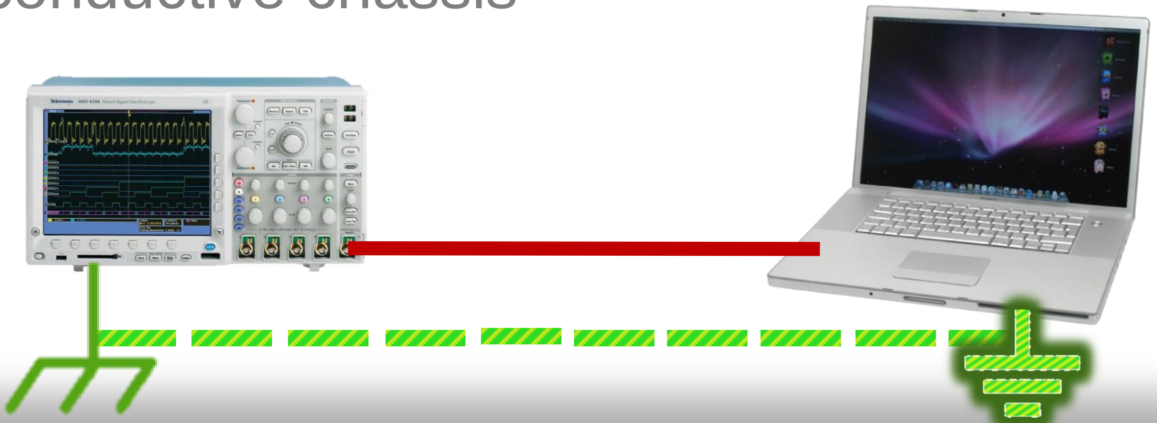


Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis



Ground-potential analysis

- **Attenuating EMI emanations**
  "Unwanted currents or electromagnetic fields? Dump them to the circuit ground!"
  (Bypass capacitors, RF shields, …)

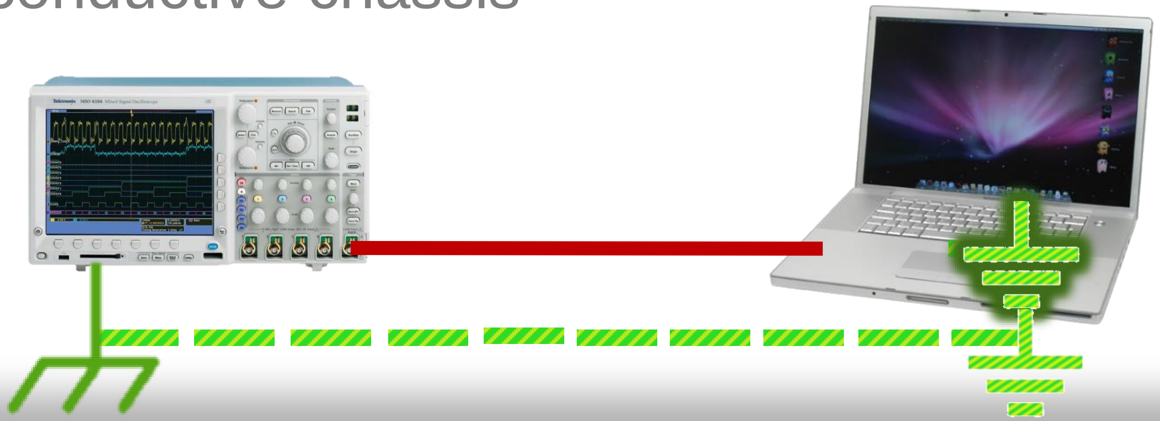
- Device is grounded, but its "ground" potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

connected to conductive chassis



Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

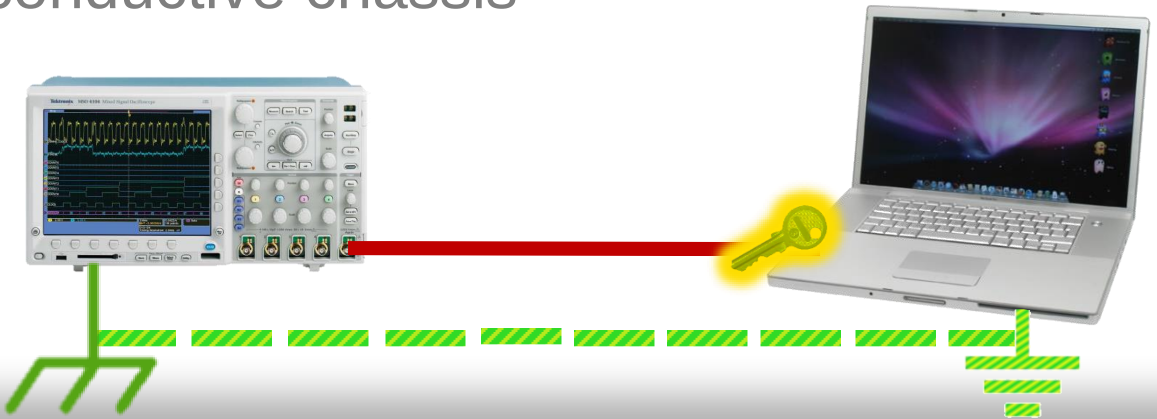
- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

connected to conductive chassis

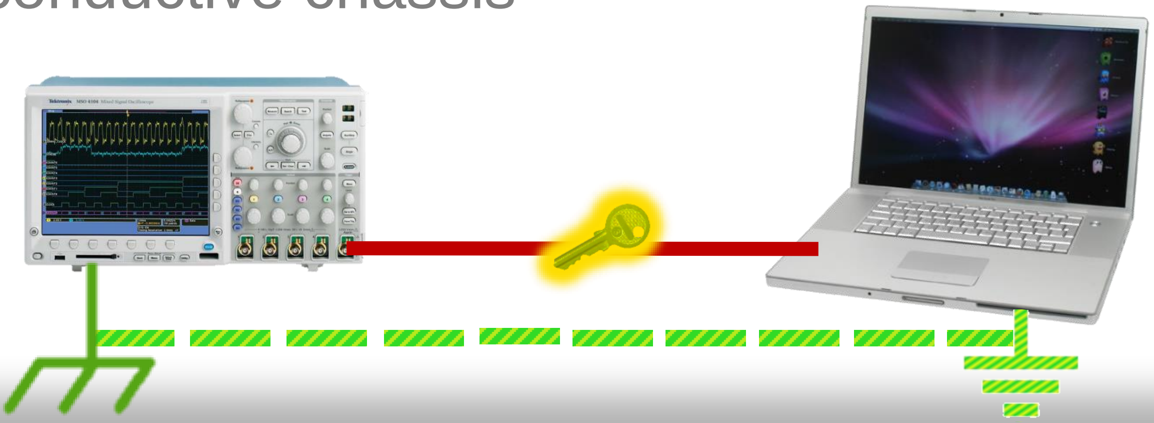


Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis

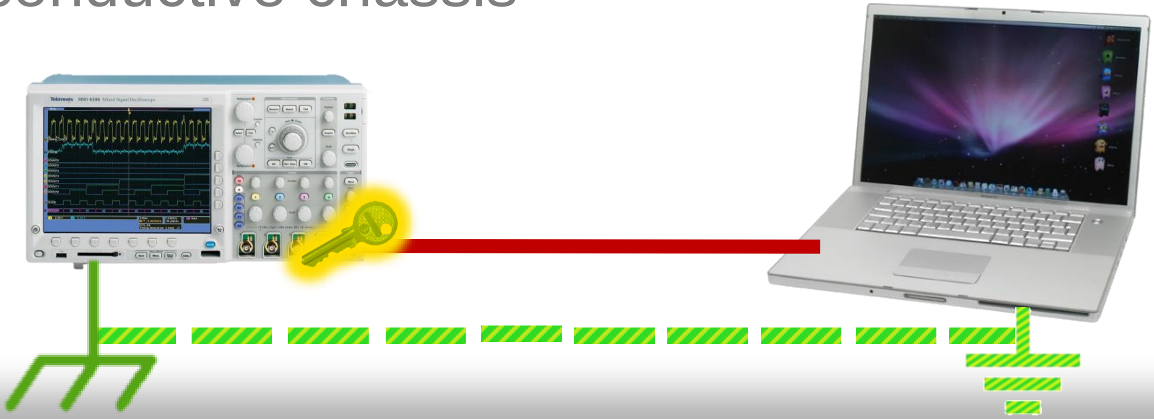


Ground-potential analysis

- **Attenuating EMI emanations**
  "Unwanted currents or electromagnetic fields? Dump them to the circuit ground!"
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its "ground" potential fluctuates relative to the mains earth ground.

	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis

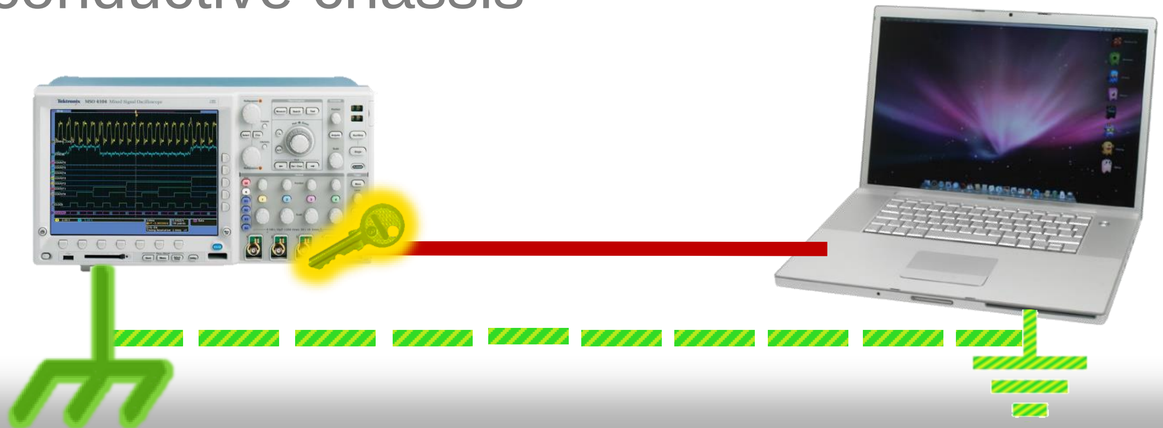


Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

	Computation
affects	currents and EM fields
dumped to	device ground
connected to	conductive chassis



Ground-potential analysis

- **Attenuating EMI emanations**
  "Unwanted currents or electromagnetic fields? Dump them to the circuit ground!"
  (Bypass capacitors, RF shields, …)

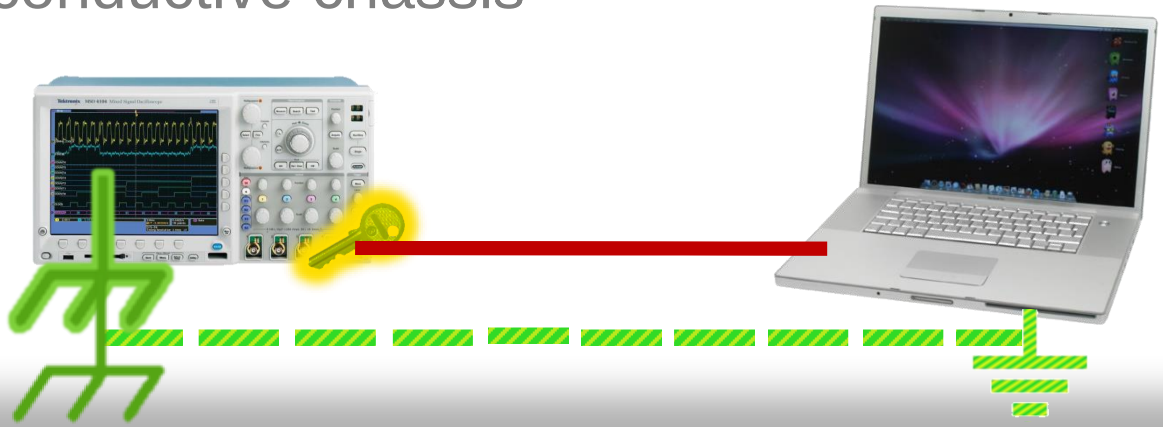
- Device is grounded, but its "ground" potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

connected to conductive chassis



Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

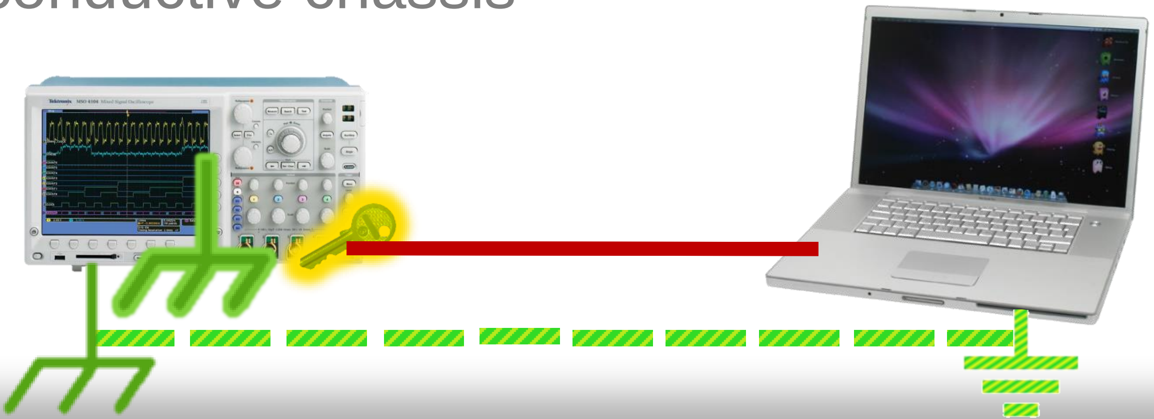
- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

connected to conductive chassis



Ground-potential analysis

- **Attenuating EMI emanations**
  “Unwanted currents or electromagnetic fields? Dump them to the circuit ground!”
  (Bypass capacitors, RF shields, …)

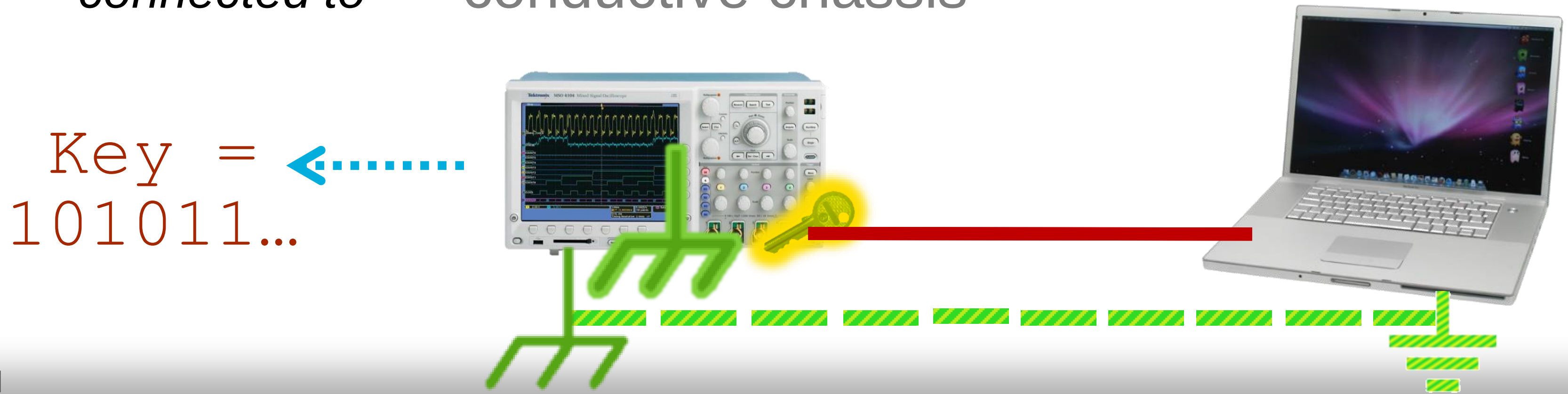
- Device is grounded, but its “ground” potential fluctuates relative to the mains earth ground.

Computation

affects currents and EM fields

dumped to device ground

connected to conductive chassis



```
Key =
101011…
```

Our results

- Channels for attacking PCs
  - Ground potential (chassis and others)
  - Power
  - Electromagnetic

Our results

- Channels for attacking PCs
  - Ground potential (chassis and others)
  - Power
  - Electromagnetic

- Exploited via low-bandwidth cryptanalytic attacks
  - Adaptive attack (50 kHz bandwidth) [Genkin Shamir Tromer 14]
  - Non-adaptive attack (1.5 MHz bandwidth)

Our results

- Channels for attacking PCs
  - Ground potential (chassis and others)
  - Power
  - Electromagnetic

- Exploited via low-bandwidth cryptanalytic attacks
  - Adaptive attack (50 kHz bandwidth) [Genkin Shamir Tromer 14]
  - Non-adaptive attack (1.5 MHz bandwidth)

- Common cryptographic software
  - GnuPG 1.4.15 (CVE 2013-4576, CVE-2014-5270)
  - RSA, ElGamal
  - Worked with GnuPG developers to mitigate the attack

Our results

- Channels for attacking PCs
  - Ground potential (chassis and others)
  - Power
  - Electromagnetic

- Exploited via low-bandwidth cryptanalytic attacks
  - Adaptive attack (50 kHz bandwidth) [Genkin Shamir Tromer 14]
  - Non-adaptive attack (1.5 MHz bandwidth)

- Common cryptographic software
  - GnuPG 1.4.15 (CVE 2013-4576, CVE-2014-5270)
  - RSA, ElGamal
  - Worked with GnuPG developers to mitigate the attack

GnuPG

- Applicable to various laptop models

Our results

- Channels for attacking PCs
  - Ground potential (chassis and others)
  - Power
  - Electromagnetic

- Exploited via low-bandwidth cryptanalytic attacks
  - Adaptive attack (50 kHz bandwidth) [Genkin Shamir Tromer 14]
  - Non-adaptive attack (1.5 MHz bandwidth) **today**

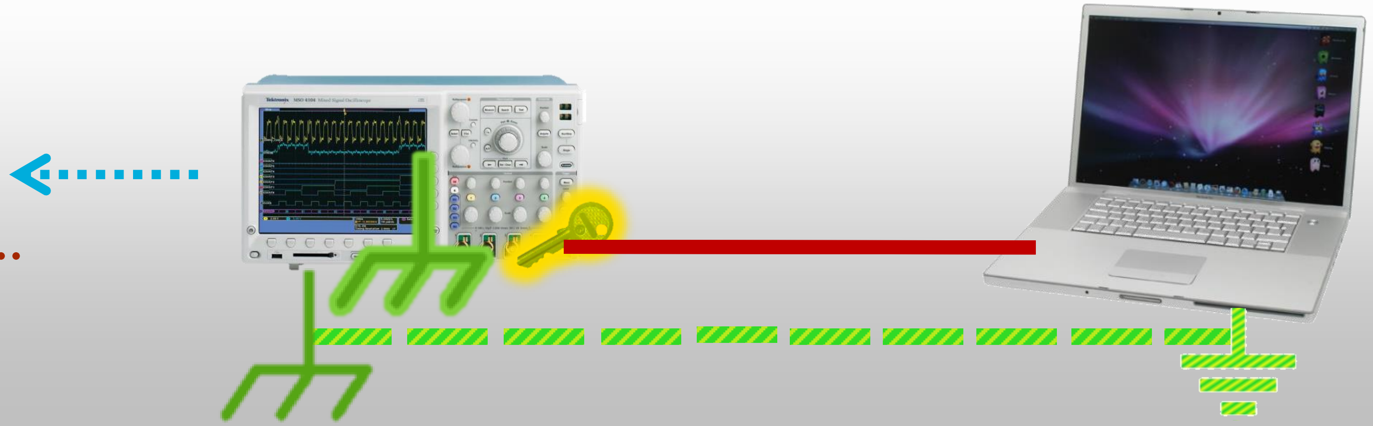
- Common cryptographic software
  - GnuPG 1.4.15 (CVE 2013-4576, CVE-2014-5270)
  - RSA, ElGamal
  - Worked with GnuPG developers to mitigate the attack

GnuPG

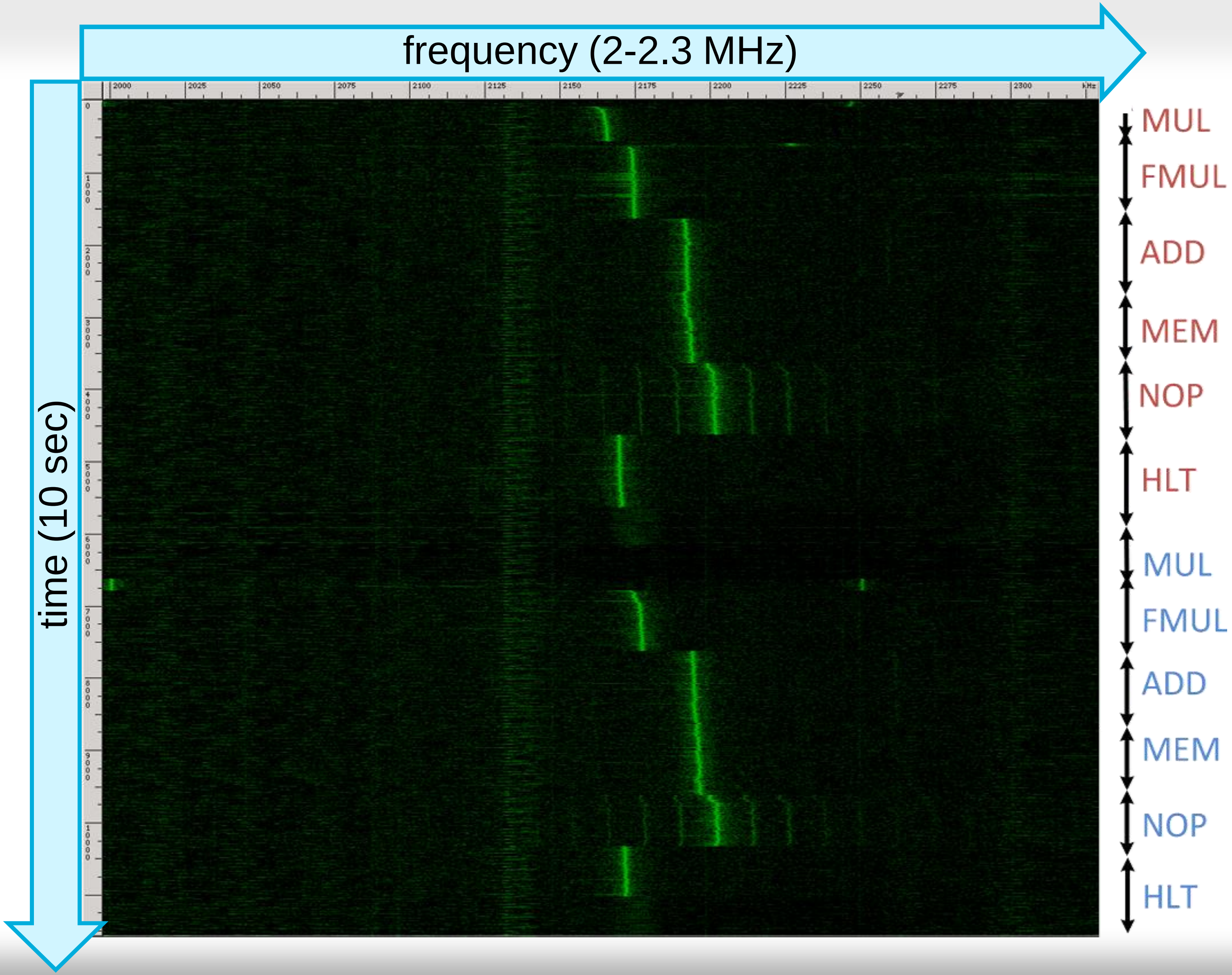
- Applicable to various laptop models

Demo:
distinguishing instructions

Key =
101011…



Distinguishing various CPU operations



Low-bandwidth leakage of RSA

Definitions (RSA)

Key setup

- **sk:** random primes p, q, private exponent d
- **pk:** $n = pq$, public exponent e

Encryption

$$c = m^e \, mod \, n$$

Decryption

$$m = c^d \, mod \, n$$

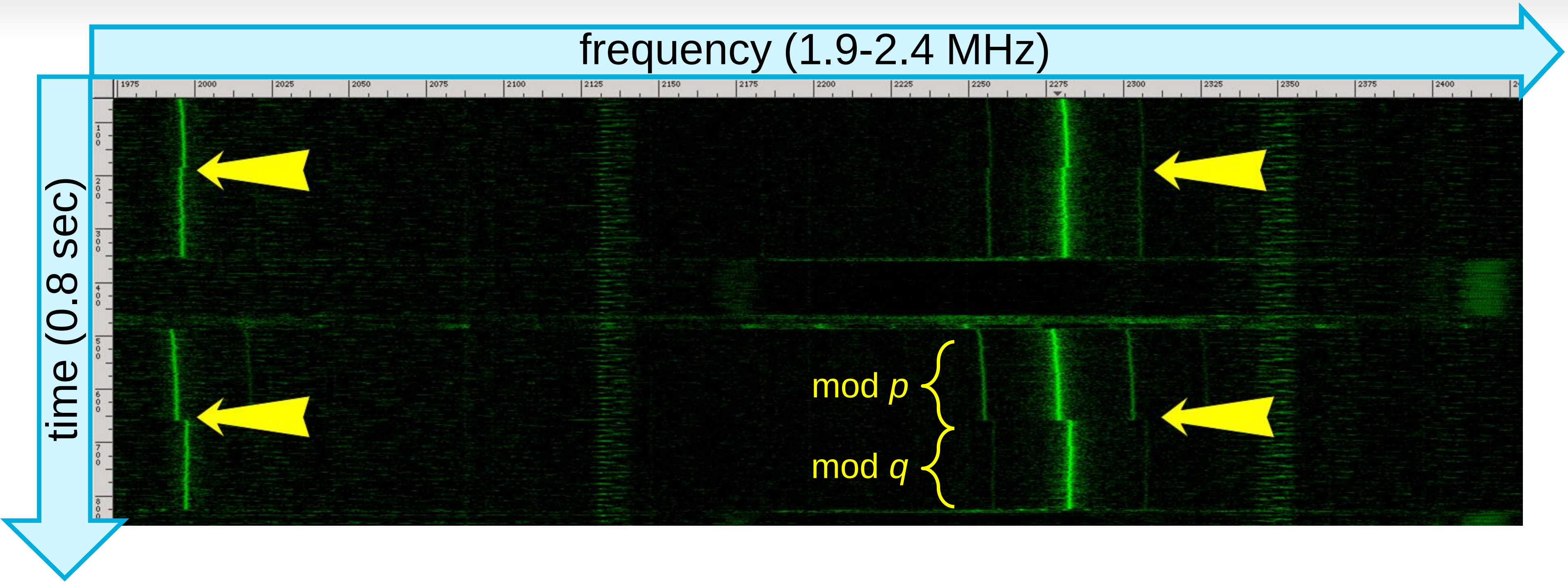
A quicker way used by most implementations

$$m_p = c^{d_p} \, mod \, p$$

$$m_q = c^{d_q} \, mod \, q$$

Obtain m using Chinese Remainder Theorem

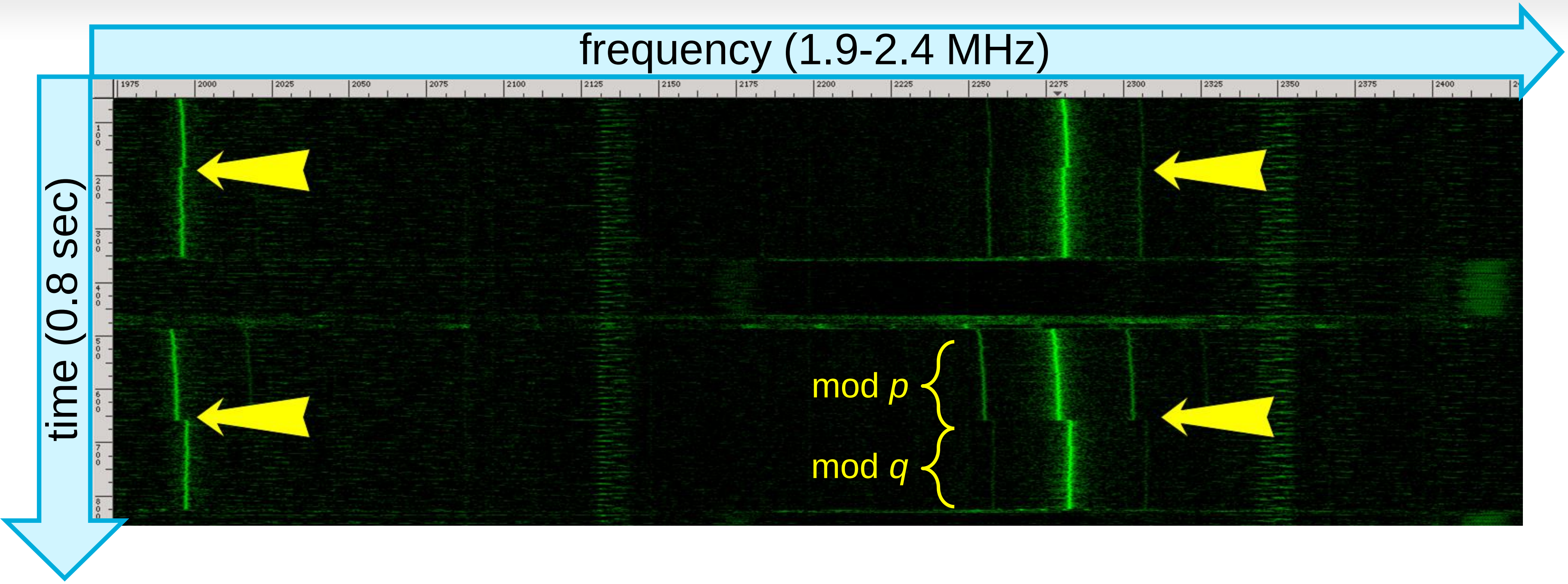
GnuPG RSA key distinguishability



Can distinguish between:

1. Decryptions and other operations

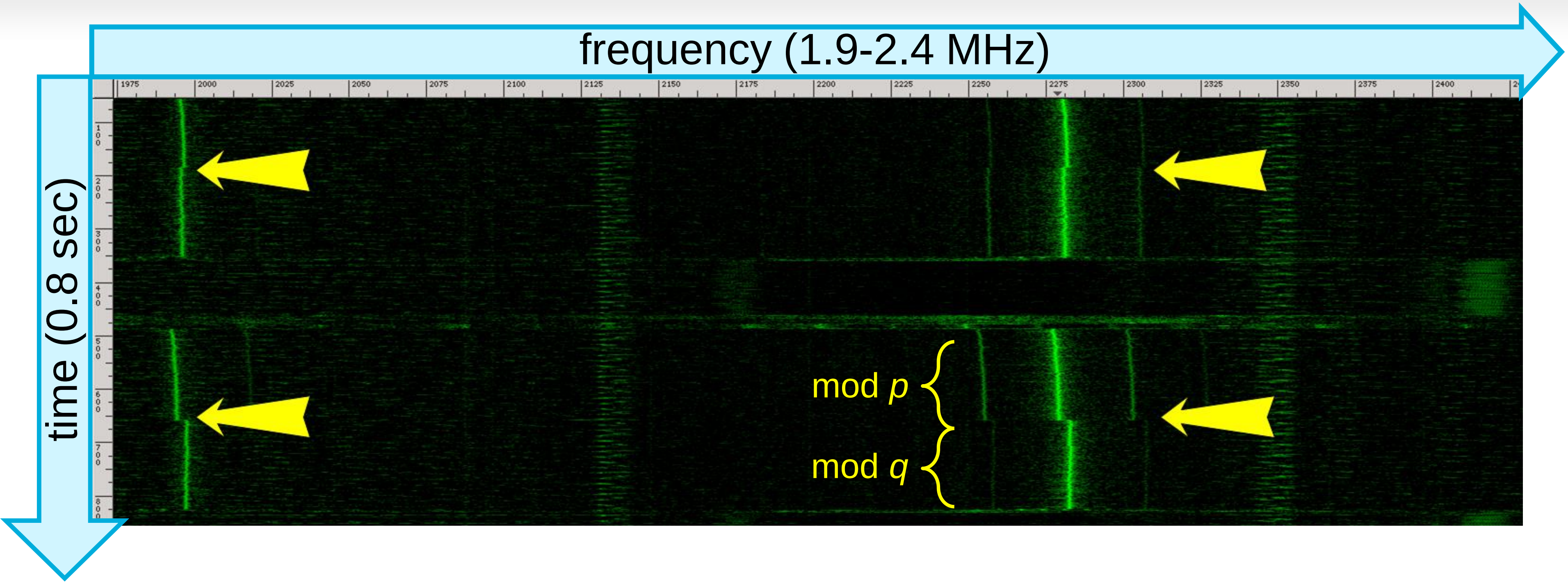
GnuPG RSA key distinguishability



Can distinguish between:

1. Decryptions and other operations
2. Two exponentiations (mod p, mod q)

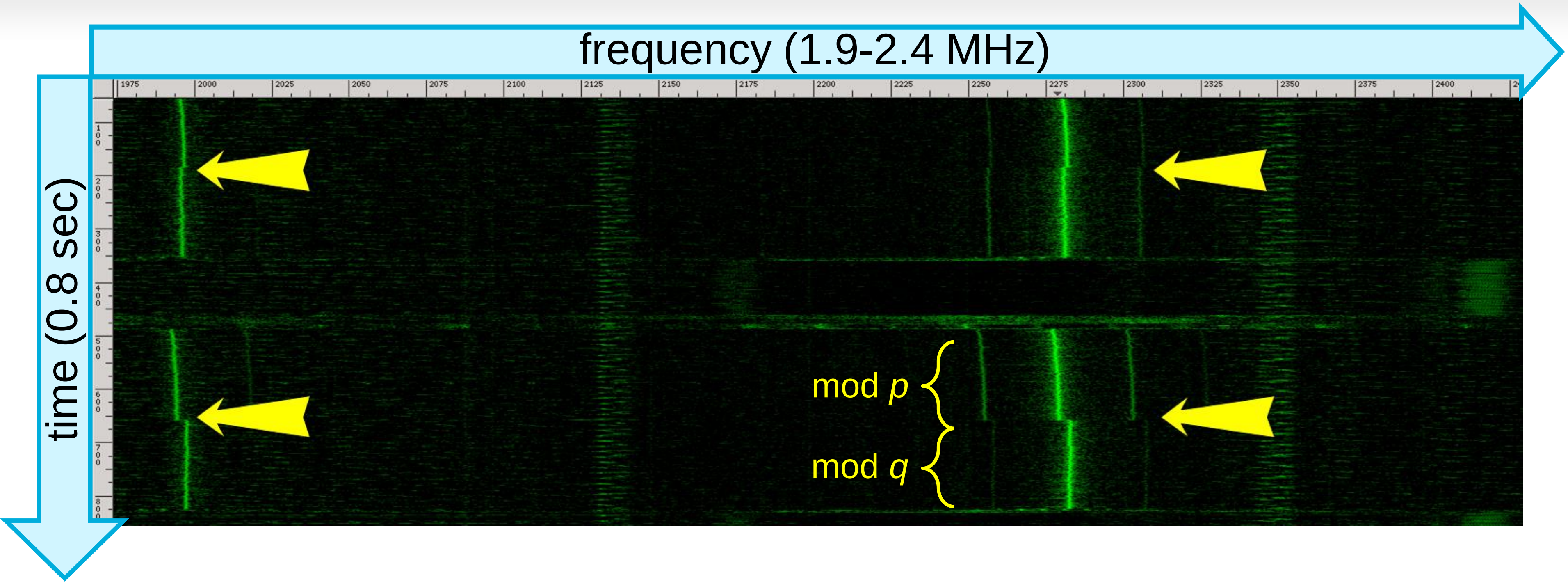
GnuPG RSA key distinguishability



Can distinguish between:

1. Decryptions and other operations
2. Two exponentiations (mod p, mod q)
3. Different keys

GnuPG RSA key distinguishability



Can distinguish between:

1. Decryptions and other operations
2. Two exponentiations (mod p, mod q)
3. Different keys
4. Different primes

Key extraction

Amplifying the key dependency

- **Difficulties when attacking RSA**
  - **2GHz CPU speed vs. 1.5MHz measurements**
  - **Cannot rely on a single key-dependent instruction**

Amplifying the key dependency

- **Difficulties when attacking RSA**
  - **2GHz CPU speed vs. 1.5MHz measurements**
  - **Cannot rely on a single key-dependent instruction**

- **Idea: leakage self-amplification** [Genkin Shamir Tromer 2014]

  abuse algorithm’s own code to amplify its own leakage!

Amplifying the key dependency

- **Difficulties when attacking RSA**
  - **2GHz CPU speed vs. 1.5MHz measurements**
  - **Cannot rely on a single key-dependent instruction**

- **Idea: <u>leakage self-amplification</u>** [Genkin Shamir Tromer 2014]

  abuse algorithm’s own code to amplify its own leakage!
  - **Craft suitable cipher-text to affect the inner-most loop**

Amplifying the key dependency

- **Difficulties when attacking RSA**
  - **2GHz CPU speed vs. 1.5MHz measurements**
  - **Cannot rely on a single key-dependent instruction**

- **Idea: leakage self-amplification** [Genkin Shamir Tromer 2014]
  abuse algorithm's own code to amplify its own leakage!
  - **Craft suitable cipher-text to affect the inner-most loop**
  - **Small differences in repeated inner-most loops cause a big overall difference in code behavior**

Amplifying the key dependency

- **Difficulties when attacking RSA**
  - **2GHz CPU speed vs. 1.5MHz measurements**
  - **Cannot rely on a single key-dependent instruction**

- **Idea: leakage self-amplification** [Genkin Shamir Tromer 2014]
  abuse algorithm's own code to amplify its own leakage!
  - **Craft suitable cipher-text to affect the inner-most loop**
  - **Small differences in repeated inner-most loops cause a big overall difference in code behavior**
  - **Measure low-bandwidth leakage**

GnuPG modular exponentiation

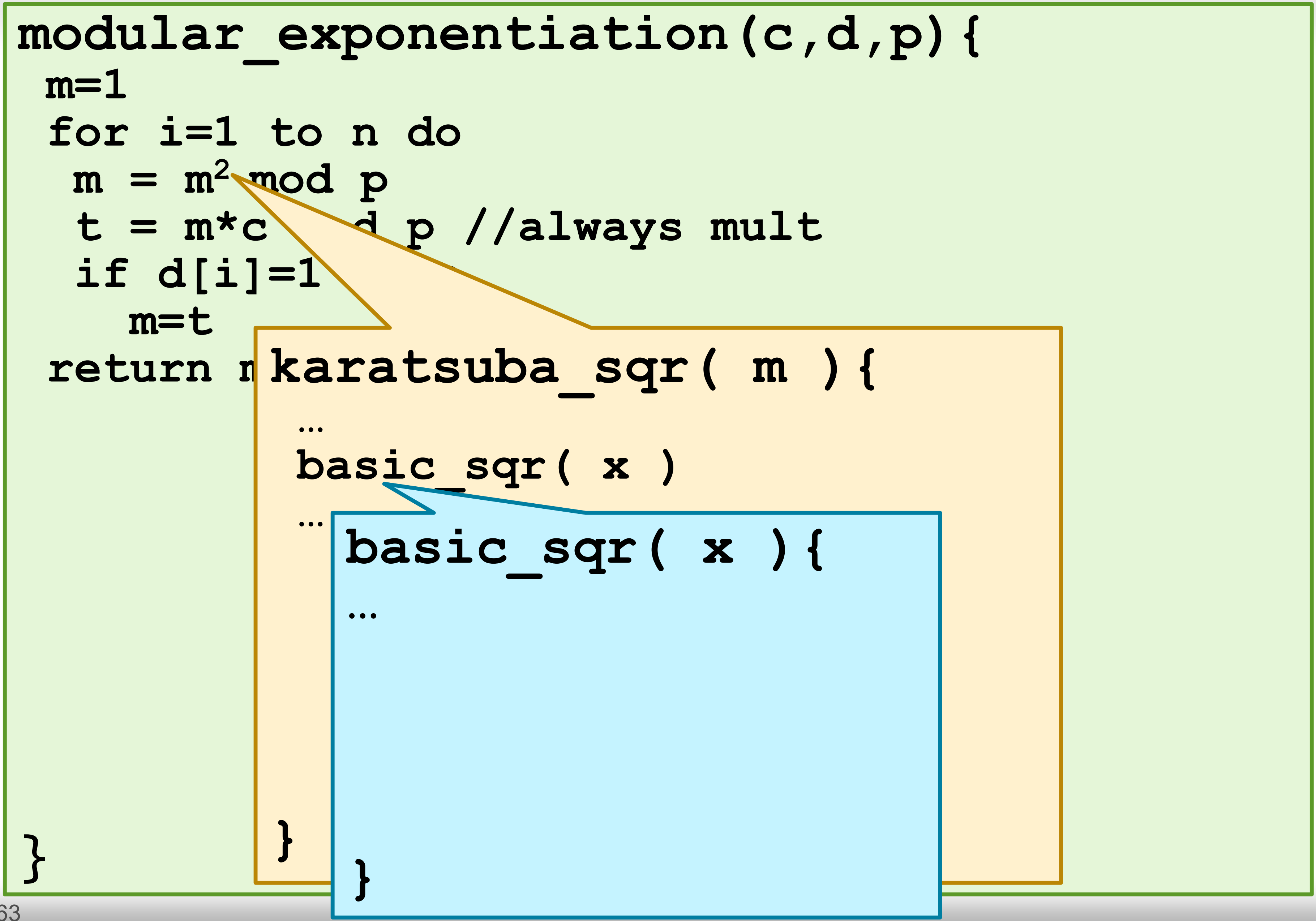
```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = $m^2$ mod p
  t = m*c mod p //always mult
  if d[i]=1 then
    m=t
 return m
}
```

GnuPG modular exponentiation

```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = $m^2$ mod p
  t = m*c    d p //always mult
  if d[i]=1
    m=t
 return m
}
```

```
karatsuba_sqr( m ){
 …
 basic_sqr( x )
 …
}
```

GnuPG modular exponentiation



```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = m^2 mod p
  t = m*c mod p //always mult
  if d[i]=1
    m=t
 return m

}
```

```
karatsuba_sqr( m ){
  …
  basic_sqr( x )
  …

}
```

```
basic_sqr( x ){
…


}
```

GnuPG modular exponentiation

```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = m$^2$ mod p
  t = m*c mod p //always mult
  if d[i]=1
    m=t
 return m

}
```

```
karatsuba_sqr( m ){
 …
 basic_sqr( x )
 …

}
```

```
basic_sqr( x ){
…
```

```
if( x[j]==0)
 y = 0
else
 y = x[j]*x
```

```
}
```

GnuPG modular exponentiation

```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = m^2 mod p
  t = m*c    d p //always mult
  if d[i]=1
    m=t
 return m
}
```

```
karatsuba_sqr( m ){
  …
  basic_sqr( x )
  …
}
```

```
basic_sqr( x ){
…
 if( x[j]==0)
  y = 0
 else
  y = x[j]*x
}
```

x7

GnuPG modular exponentiation

```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = m^2 mod p
  t = m*c   d p //always mult
  if d[i]=1
    m=t
 return m
}
```

```
karatsuba_sqr( m ){
  …
  basic_sqr( x )
  …
}
```

x27

```
basic_sqr( x ){
…
  if( x[j]==0)
   y = 0
  else
   y = x[j]*x
}
```

x7

GnuPG modular exponentiation

```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = $m^2$ mod p
  t = m*c    d p //always mult
  if d[i]=1
    m=t
 return m
}
```

```
karatsuba_sqr( m ){
  …
  basic_sqr( x )
  …
}
```

x27

```
basic_sqr( x ){
…
 if( x[j]==0)
  y = 0
 else
  y = x[j]*x
}
```

x7

repeated 189 times per bit of d

~0.2ms of measurement per bit of d

GnuPG modular exponentiation

```
modular_exponentiation(c,d,p){
 m=1
 for i=1 to n do
  m = m^2 mod p
  t = m*c    d p //always mult
  if d[i]=1
    m=t
 return r
}
```

```
karatsuba_sqr( m ){
  …
  basic_sqr( x )
  …
}
```

```
basic_sqr( x ){
…
 if( x[j]==0)
  y = 0
 else
  y = x[j]*x
}
```

x27

x7

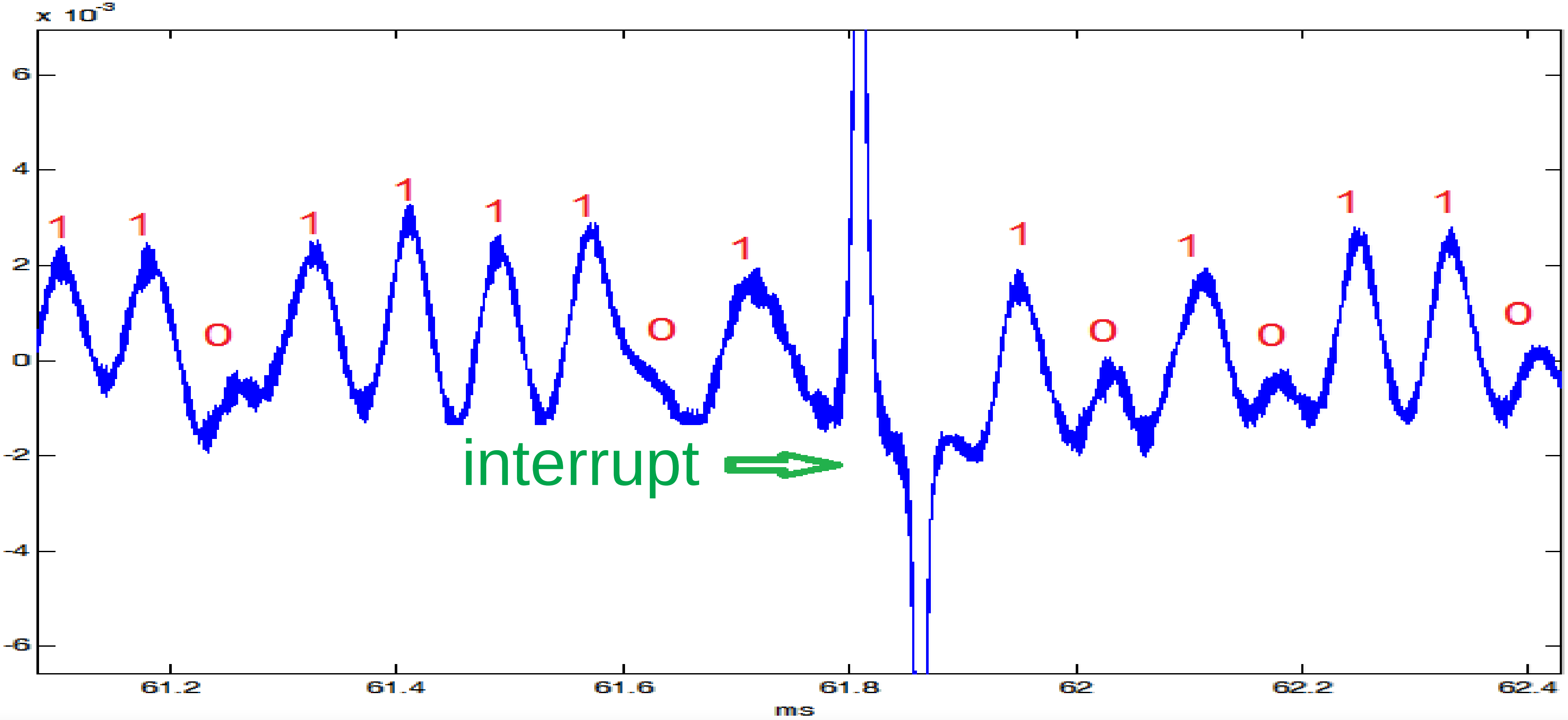
repeated 189 times per bit of d

~0.2ms of measurement per bit of d

craft c such that
$d[i] = 1 \rightarrow x[j] = 0$
$d[i] = 0 \rightarrow x[j] \neq 0$
(for most j's)

Reading the secret key (non-adaptive attack)

- Acquire trace
- Filter around carrier (1.7 MHz)
- FM demodulation
- Read out bits (“simple ground analysis”)



A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):

A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):
  - RSA: $c = N - 1$

A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):
  - RSA: $c = N - 1$
  - ElGamal: $c = p - 1$

A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):
  - RSA: $c = N - 1$
  - ElGamal: $c = p - 1$

- Total #measurements:

Attack type	# of traces	Time	Bandwidth	Cipher

A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):
  - RSA: $c = N - 1$
  - ElGamal: $c = p - 1$

- Total #measurements:

Attack type	# of traces	Time	Bandwidth	Cipher
Non-adaptive chosen ciphertext	3-15	3 sec	2 MHz	ElGamal, RSA

A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):
  - RSA: $c = N - 1$
  - ElGamal: $c = p - 1$

- Total #measurements:

Attack type	# of traces	Time	Bandwidth	Cipher
Non-adaptive chosen ciphertext	3-15	3 sec	2 MHz	ElGamal, RSA
Adaptive chosen ciphertext	2048	1 hour	50 kHz	RSA

A chosen ciphertext attack

- Non-adaptive ciphertext choice $c \equiv -1 \bmod p$ (similar to [YLMH05]):
  - RSA: $c = N - 1$
  - ElGamal: $c = p - 1$

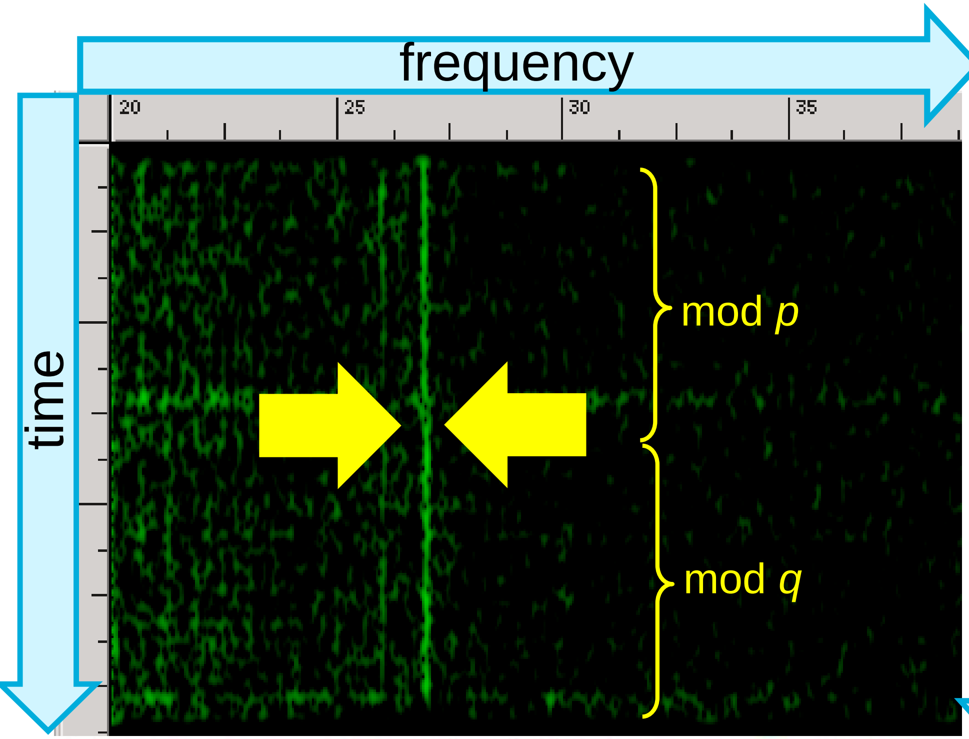
- Total #measurements:

Attack type	# of traces	Time	Bandwidth	Cipher
Non-adaptive chosen ciphertext	3-15	3 sec	2 MHz	ElGamal, RSA
Adaptive chosen ciphertext	2048	1 hour	50 kHz	RSA

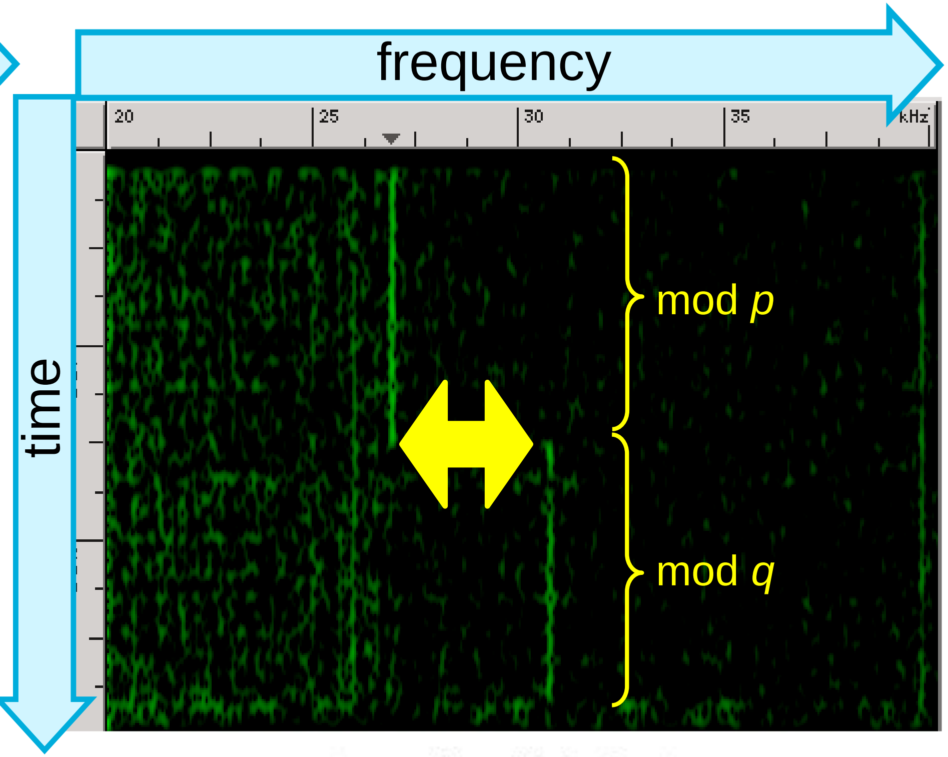
- Send chosen ciphertexts using Enigmail

Empirical results

Reading the secret key (adaptive attack)



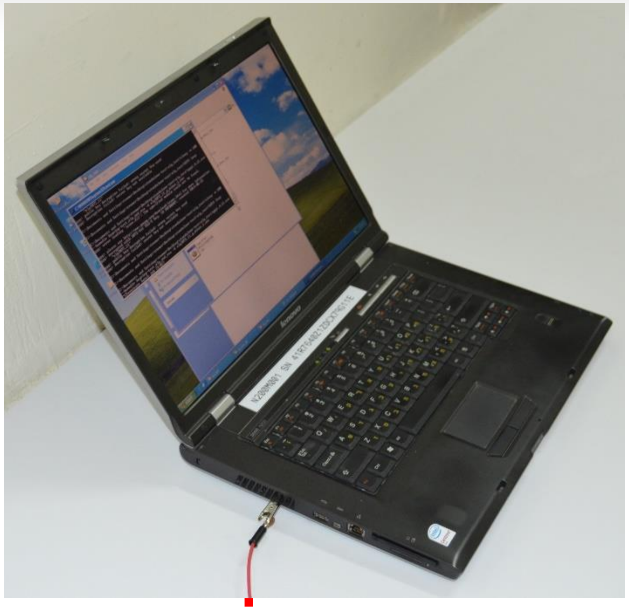
Attacked bit is 0

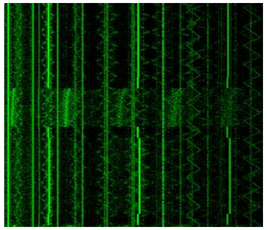


Attacked bit is 1

Demo:
key extraction

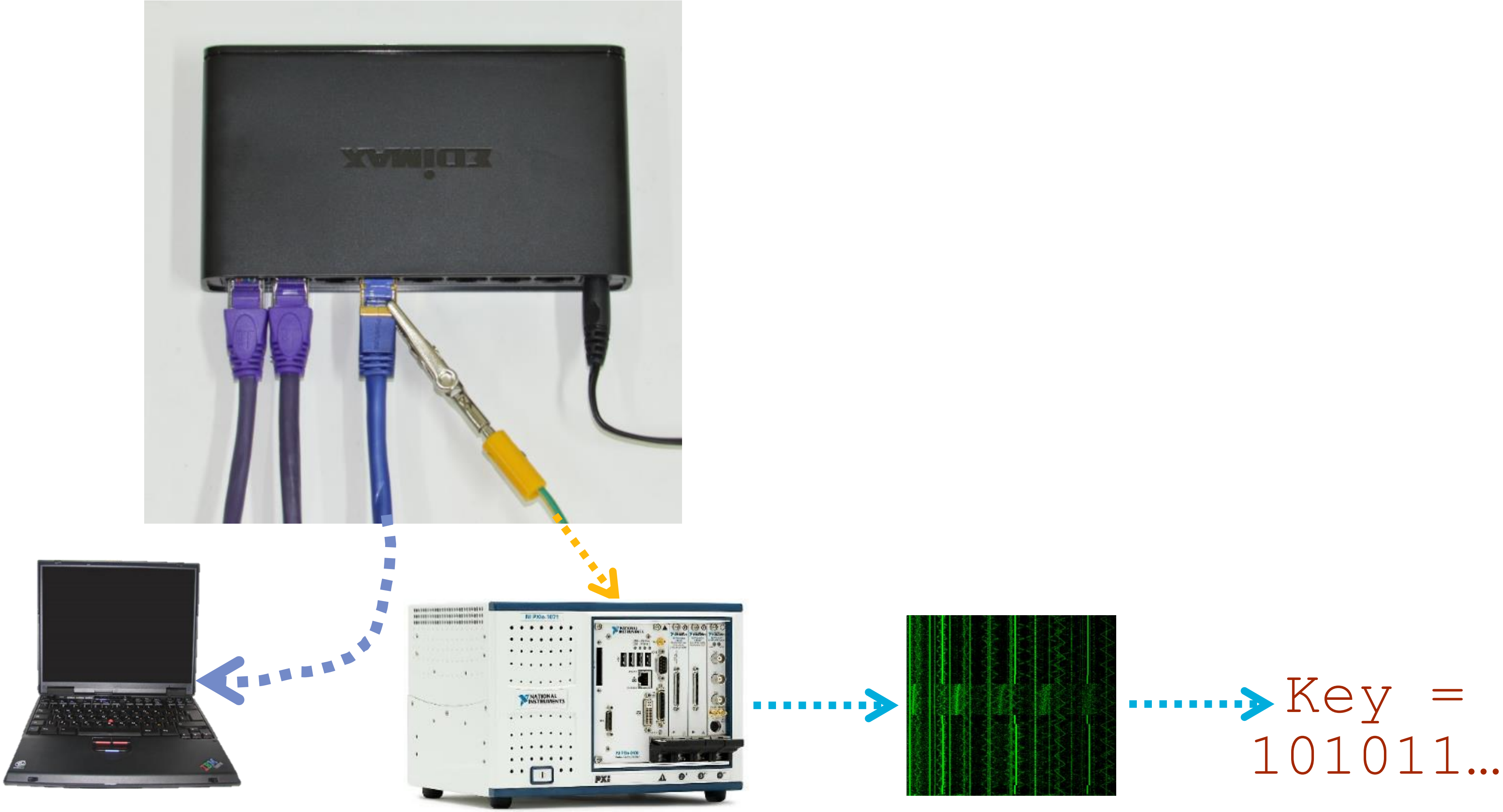
RSA and ElGamal key extraction in a few seconds using direct chassis measurement (non-adaptive attack)





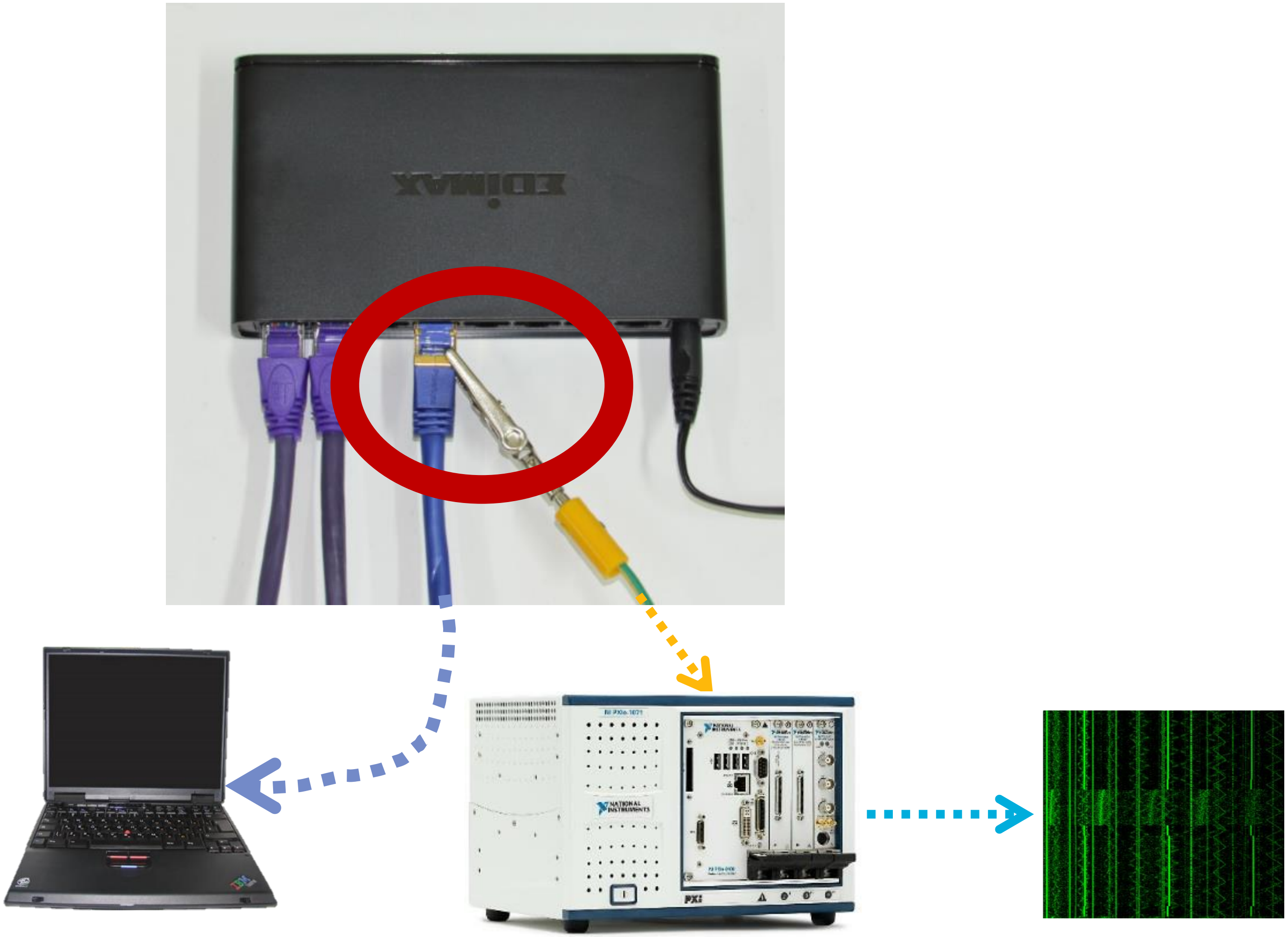
Key = 101011…

RSA and ElGamal key extraction in a few seconds using the far end of 10 meter network cable (non-adaptive attack)



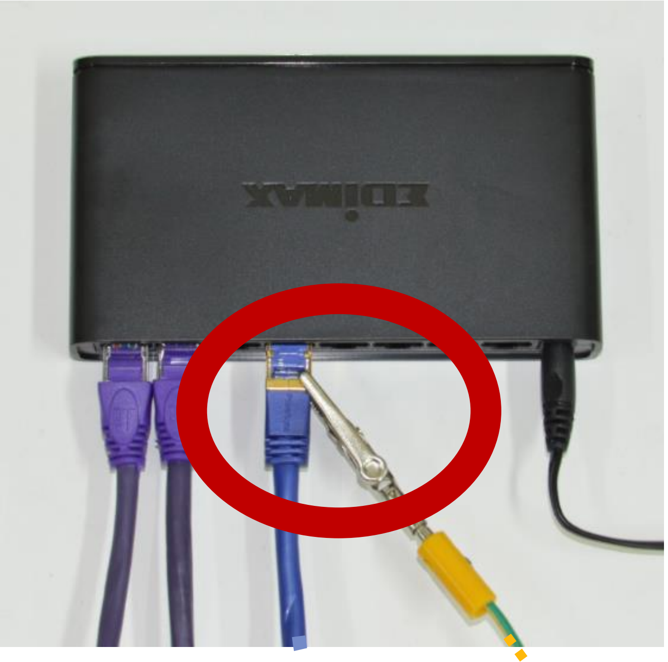
Key = 101011…

RSA and ElGamal key extraction in a few seconds using the far end of 10 meter network cable (non-adaptive attack)

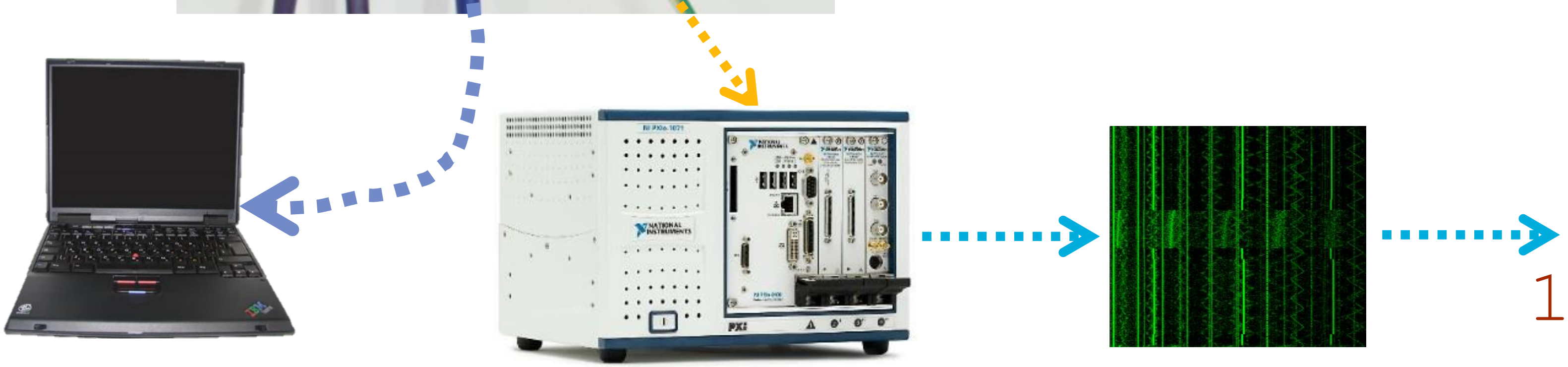


Key = 101011…

RSA and ElGamal key extraction in a few seconds using the far end of 10 meter network cable (non-adaptive attack)

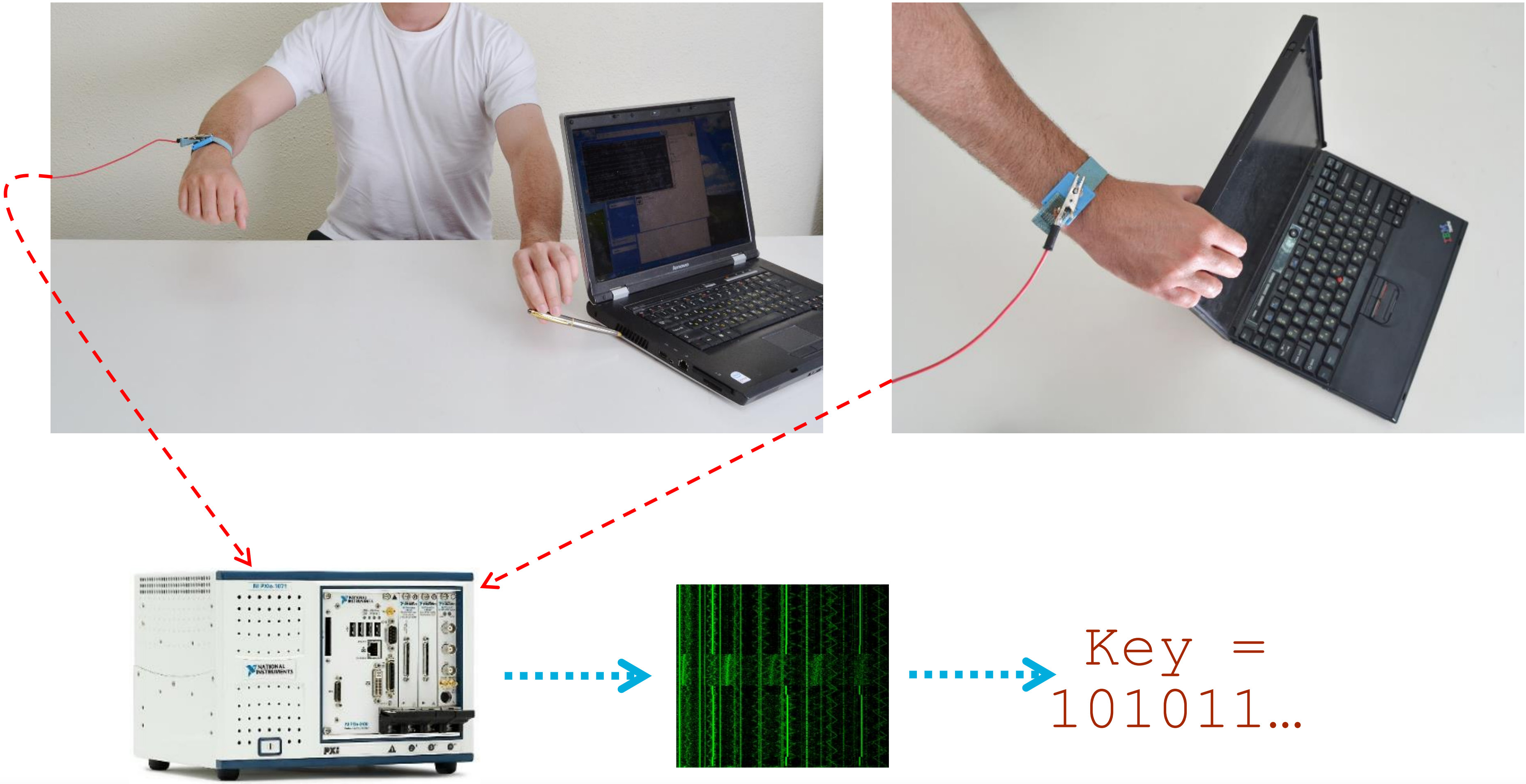


works even if a firewall is present, or port is turned off



Key = 101011…

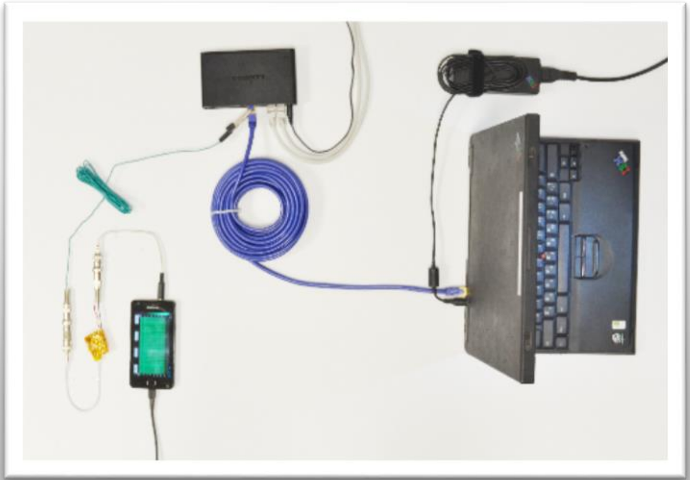
RSA and ElGamal key extraction in a few seconds using human touch (non-adaptive attack)



Key = 101011…

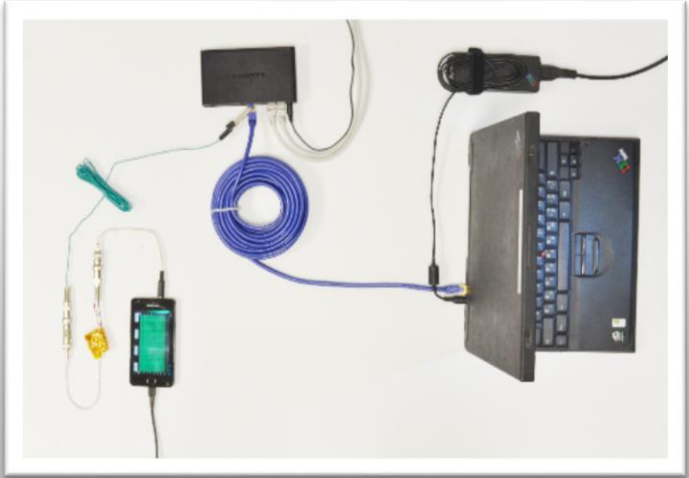
Thanks!

cs.tau.ac.il/~tromer/handsoff



Thanks!

`cs.tau.ac.il/~tromer/handsoff`



Thanks!

cs.tau.ac.il/~tromer/handsoff