LadderLeak

Breaking ECDSA with Less than One Bit of Nonce Leakage

Real World Crypto 2021 (ePrint 2020/615, appeared at CCS '20)

Diego F. Aranha[1] Felipe R. Novaes[2] Akira Takahashi[1] Mehdi Tibouchi[3] Yuval Yarom[4]

[1]DIGIT, Aarhus University, Denmark

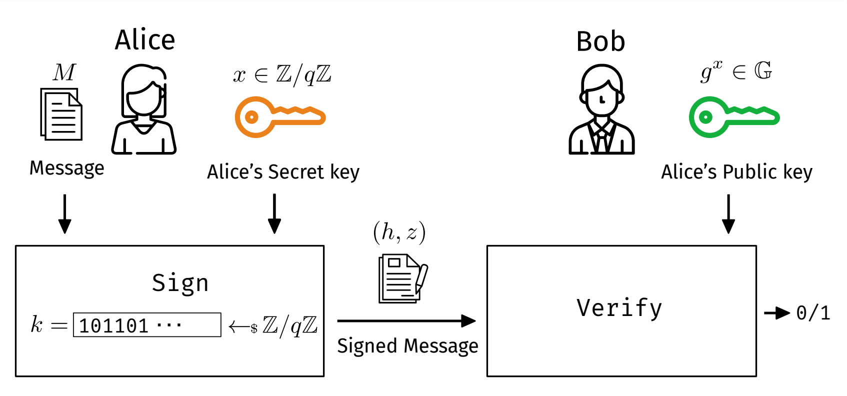
[2]University of Campinas, Brazil

[3]NTT Corporation, Japan

[4]University of Adelaide and Data61, Australia

Nonce = Number used only once

“Nonce” in ECDSA/Schnorr-type Schemes

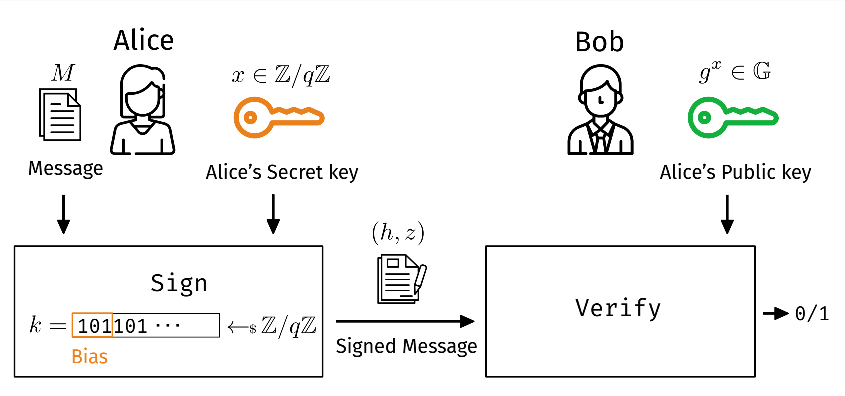


- k is a uniformly random value satisfying

$$k \equiv \underbrace{z}_{\text{public}} + \underbrace{h}_{\text{public}} \cdot x \mod q.$$

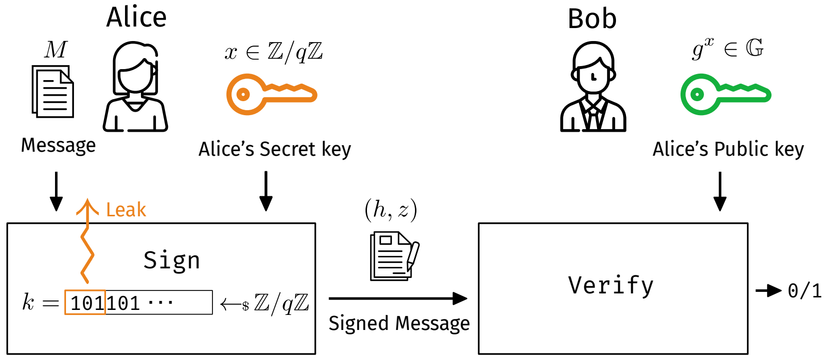
- k should **NEVER** be reused/exposed as $x = (z - z')/(h' - h) \mod q$

Risk of Biased/Leaky Nonces



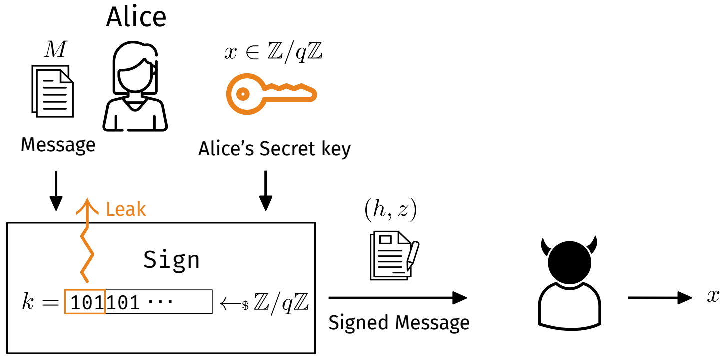
- What if k is slightly biased ?
- Secret key x is recovered by solving the **hidden number problem (HNP)**

Risk of Biased/Leaky Nonces



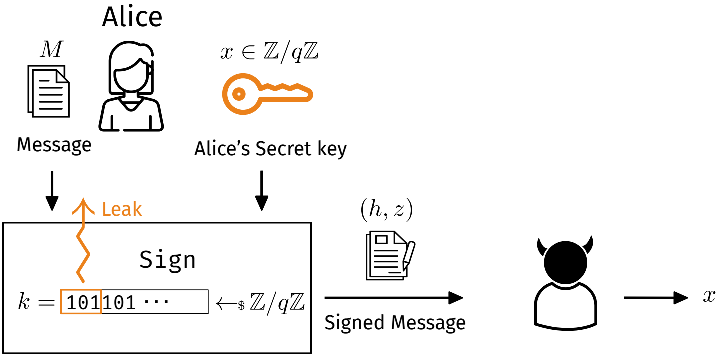
- What if k is slightly biased or partially leaked?
- Secret key x is recovered by solving the **hidden number problem (HNP)**

Risk of Biased/Leaky Nonces



- What if k is slightly biased or partially leaked? $\leadsto$ Attack!
- Secret key x is recovered by solving the **hidden number problem (HNP)**

Risk of Biased/Leaky Nonces



- What if k is slightly biased or partially leaked? $\rightsquigarrow$ Attack!
- Secret key x is recovered by solving the **hidden number problem (HNP)**

Randomness Failure in the Real World

- Poorly designed/implemented RNGs
- Predictable seed (`srand(time(0))`)
- VM resets ⤳ same snapshot will end up with the same seed
- Side-channel leakage
- and many more…

BBC | Sign in | News | Sport | Weather | Shop | Earth | Travel | Mo

NEWS

Home | UK | World | Business | Politics | Tech | Science | Health | Family & Education | E

Technology

iPhone hacker publishes secret Sony PlayStation 3 key

By Jonathan Fildes
Technology reporter, BBC News

6 January 2011

Share

The PlayStation 3's security has been broken by hackers, potentially allowing anyone to run any software - including pirated games - on the console.

A collective of hackers recently showed off a method that could force the system to reveal secret keys used to load

BBC news. 2011. https://www.bbc.com/news/technology-12116051

Randomness Failure in the Real World

- Poorly designed/implemented RNGs
- Predictable seed (`srand(time(0))`)
- VM resets ⇝ same snapshot will end up with the same seed
- Side-channel leakage
- and many more…


BBC Sign in News Sport Weather Shop Earth Travel

NEWS

Home | UK | World | Business | Politics | Tech | Science | Health | Family & Education

Technology

iPhone hacker publishes secret Sony PlayStation 3 key

By Jonathan Fildes
Technology reporter, BBC News

6 January 2011

Share

The PlayStation 3's security has been broken by hackers, potentially allowing anyone to run any software - including pirated games - on the console.

A collective of hackers recently showed off a method that could force the system to reveal secret keys used to load

BBC news. 2011. https://www.bbc.com/news/technology-12116051

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

Chronology of HNP: a 25-year retrospective

1996 Boneh–Venkatesan defined the HNP

1999 Howgrave-Graham–Smart proposed the lattice attack against HNP

2000 Bleichenbacher announced the Fourier analysis attack

⋮

2018 CacheQuote on SGX EPID; PortSmash on SMT/Hyper-Threading; ROHNP

2019 TPM-FAIL; Minerva

2020 Dé jà Vu attack on Mozilla's NSS; Raccoon attack on TLS 1.2

Still at the heart of **many** recent real-world vulnerabilities in ECDSA/Diffie–Hellman key exchange implementations!

This work

1. Improved analysis of **Fourier analysis-based attack** (Bleichenbacher '00) to solve the HNP
    - Allows us to exploit tiny amount of nonce leakage per signature
2. Novel class of cache timing attacks against the Montgomery ladder scalar multiplication in OpenSSL `1.0.2u` and `1.1.0l`, and RELIC 0.4.0.
3. Implemented a full secret key recovery attack against OpenSSL ECDSA over `sect163r1` and NIST P-192.

This work

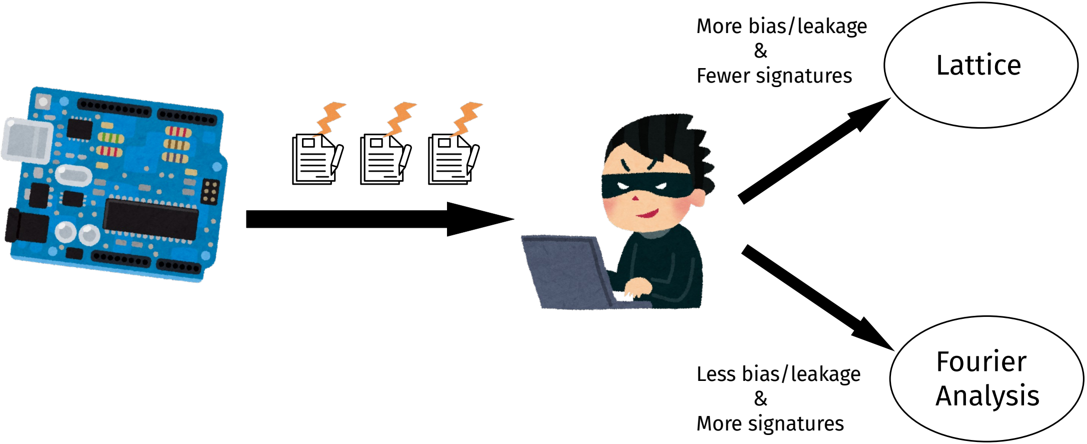
1. Improved analysis of **Fourier analysis-based attack** (Bleichenbacher '00) to solve the HNP
    - Allows us to exploit tiny amount of nonce leakage per signature
2. Novel class of cache timing attacks against the Montgomery ladder scalar multiplication in OpenSSL `1.0.2u` and `1.1.0l`, and RELIC 0.4.0.
3. Implemented a full secret key recovery attack against OpenSSL ECDSA over `sect163r1` and NIST P-192.

This work

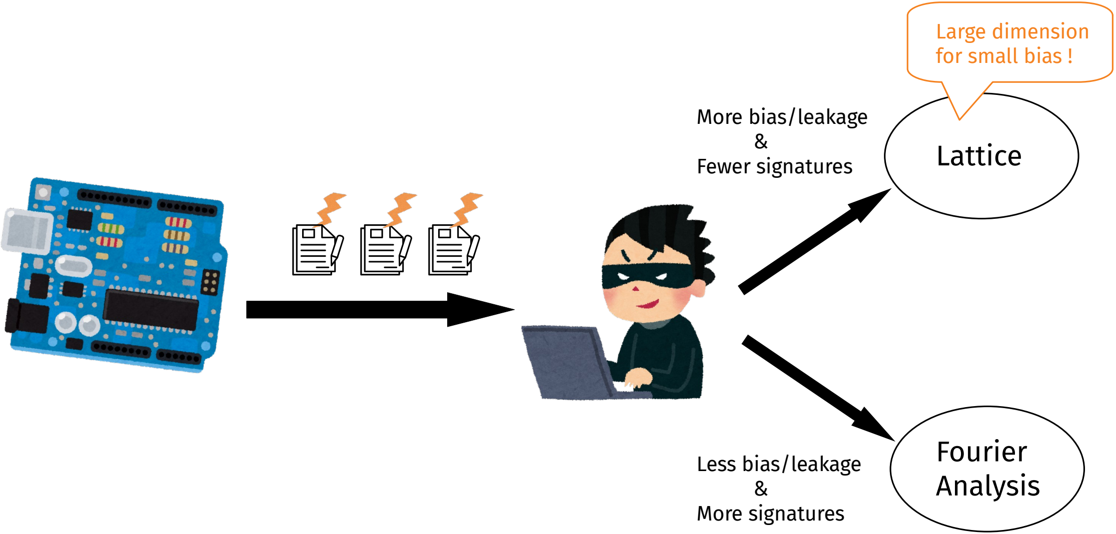
1. Improved analysis of **Fourier analysis-based attack** (Bleichenbacher '00) to solve the HNP
   - Allows us to exploit tiny amount of nonce leakage per signature
2. Novel class of cache timing attacks against the Montgomery ladder scalar multiplication in OpenSSL `1.0.2u` and `1.1.0l`, and RELIC 0.4.0.
3. Implemented a full secret key recovery attack against OpenSSL ECDSA over `sect163r1` and NIST P-192.

How to Exploit Nonce Leakage

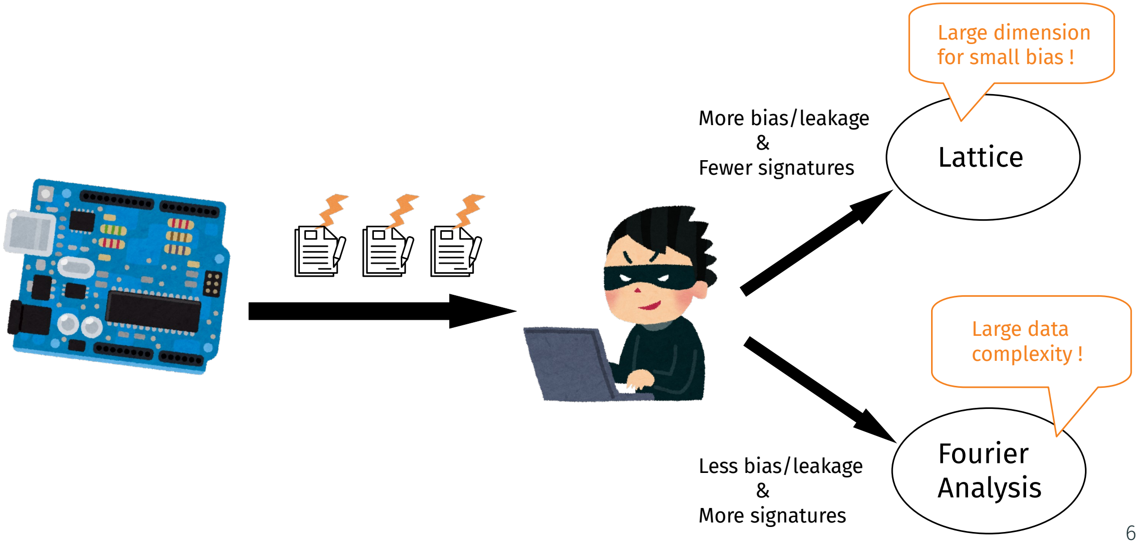
How to solve the HNP: Lattice vs Fourier analysis



How to solve the HNP: Lattice vs Fourier analysis



How to solve the HNP: Lattice vs Fourier analysis



Questions

- Can we reduce #signatures for Fourier analysis-based attack?
- Can we attack even **less than 1-bit of nonce leakage** (= MSB is only leaked with prob. < 1)?

YES!

Questions

- Can we reduce #signatures for Fourier analysis-based attack?
- Can we attack even **less than 1-bit of nonce leakage** (= MSB is only leaked with prob. < 1)?

YES!

Questions

- Can we reduce #signatures for Fourier analysis-based attack?
- Can we attack even **less than 1-bit of nonce leakage** (= MSB is only leaked with prob. < 1)?

YES!

Bleichenbacher's Attack: High-level Overview

- Step 1. Quantify the bias of nonce $K = \{k_i\}_{i \in \{1,\ldots,M\}}$
  - $\mathrm{Bias}_q(K) \approx 0$ if k is uniform in $\mathbb{Z}_q$
  - $\mathrm{Bias}_q(K) \approx 1$ if k is biased in $\mathbb{Z}_q$
  - Contribution 1: Analyzed the behavior $\mathrm{Bias}_q(K)$ when k's MSB is biased with probability < 1!
- Step 2. Find a candidate secret key which leads to the peak of $\mathrm{Bias}_q(K)$ (by computing FFT)
- Critical intermediate step: find **many small linear combinations** of integers h
  - Detect the bias peak correctly and efficiently
  - Contribution 2: Established time-data tradeoffs by applying algorithms for the **generalized birthday problem (GBP)**!

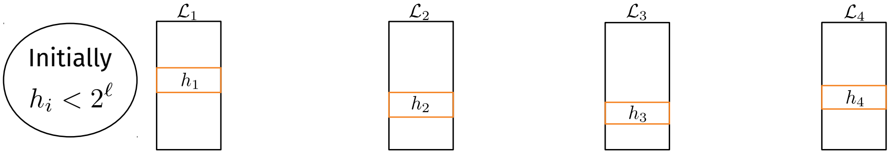
Bleichenbacher's Attack: High-level Overview

- Step 1. Quantify the bias of nonce $K = \{k_i\}_{i \in \{1,\ldots,M\}}$
  - $\text{Bias}_q(K) \approx 0$ if k is uniform in $\mathbb{Z}_q$
  - $\text{Bias}_q(K) \approx 1$ if k is biased in $\mathbb{Z}_q$
  - Contribution 1: Analyzed the behavior $\text{Bias}_q(K)$ when k's MSB is biased with probability < 1!
- Step 2. Find a candidate secret key which leads to the peak of $\text{Bias}_q(K)$ (by computing FFT)
- Critical intermediate step: find **many small linear combinations** of integers h
  - Detect the bias peak correctly and efficiently
  - Contribution 2: Established time-data tradeoffs by applying algorithms for the **generalized birthday problem (GBP)**!

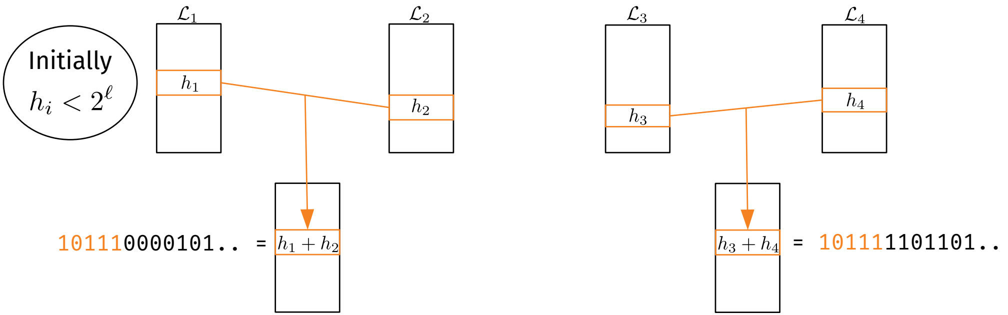
Bleichenbacher's Attack: High-level Overview

- Step 1. Quantify the bias of nonce $K = \{k_i\}_{i \in \{1,\ldots,M\}}$
  - $\text{Bias}_q(K) \approx 0$ if k is uniform in $\mathbb{Z}_q$
  - $\text{Bias}_q(K) \approx 1$ if k is biased in $\mathbb{Z}_q$
  - Contribution 1: Analyzed the behavior $\text{Bias}_q(K)$ when k's MSB is biased with probability < 1!
- Step 2. Find a candidate secret key which leads to the peak of $\text{Bias}_q(K)$ (by computing FFT)
- Critical intermediate step: find **many small linear combinations** of integers h
  - Detect the bias peak correctly and efficiently
  - Contribution 2: Established time-data tradeoffs by applying algorithms for the **generalized birthday problem (GBP)**!

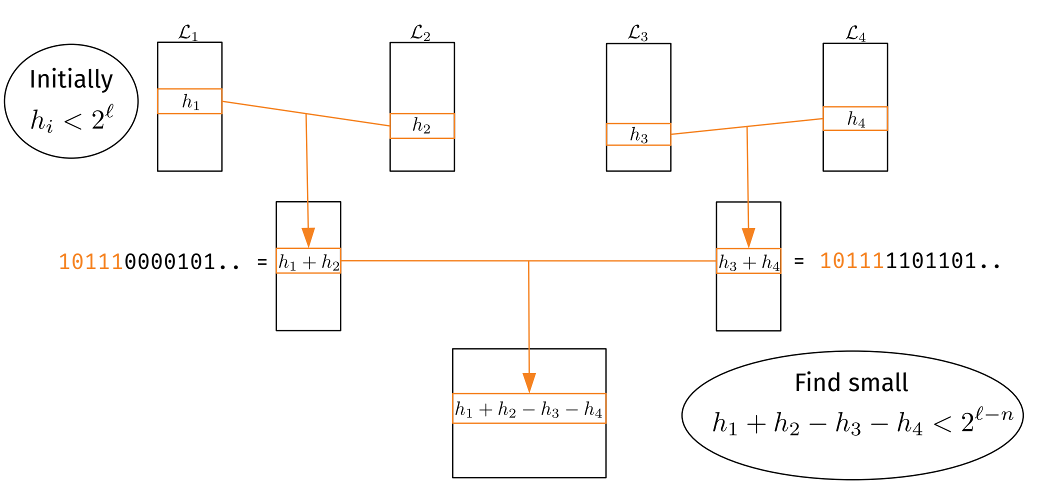
$\mathcal{K}$-list Sum for GBP (e.g., $\mathcal{K} = 4$)



$\mathcal{K}$-list Sum for GBP (e.g., $\mathcal{K} = 4$)



$\mathcal{K}$-list Sum for GBP (e.g., $\mathcal{K} = 4$)



Time–Data tradeoffs for 1-bit leakage

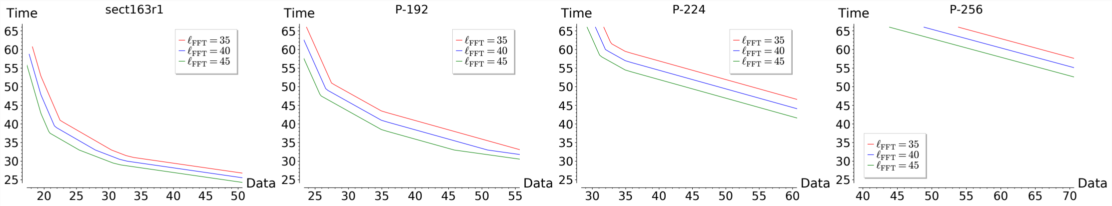


Figure 1: Time–Data tradeoff graphs (in a $\log_2$ scale) when memory is fixed to 2^{35}

* Optimized data complexity by solving the linear programming problem
* Further optimization is feasible if > 1-bit leakage is available!
    - Sample amplification via exhaustive $\mathcal{K}$-sum search

ECDSA key recovery attack: experimental records

Target	Bias	Facility	Error rate	Input	Thread (Collision)	Time (Collision)	RAM (Collision)	Recovered MSBs
NIST P-192	1-bit	AWS EC2	0	2^{29}	96×24	113h	492GB	39
NIST P-192	1-bit	AWS EC2	1%	2^{35}	96×24	52h	492GB	39
`sect163r1`	1-bit	Cluster	0	2^{23}	16×16	7h	80GB	36
`sect163r1`	1-bit	Workstation	2.7%	2^{24}	48	42h	250GB	35
`sect163r1`	2-bit	Cluster	0	1024	16	2h	96GB	32

Table 1: Computational results for the first round of Bleichenbacher

- Attack on **P-192** is made possible by our highly optimized parallel implementation.
- Attack on `sect163r1` is even feasible with a laptop.
- Recovering remaining bits is much cheaper in Bleichenbacher's framework.

ECDSA key recovery attack: experimental records

Target	Bias	Facility	Error rate	Input	Thread (Collision)	Time (Collision)	RAM (Collision)	Recovered MSBs
NIST P-192	1-bit	AWS EC2	0	2^{29}	96×24	113h	492GB	39
NIST P-192	1-bit	AWS EC2	1%	2^{35}	96×24	52h	492GB	39
`sect163r1`	1-bit	Cluster	0	2^{23}	16×16	7h	80GB	36
`sect163r1`	1-bit	Workstation	2.7%	2^{24}	48	42h	250GB	35
`sect163r1`	2-bit	Cluster	0	1024	16	2h	96GB	32

Table 1: Computational results for the first round of Bleichenbacher

- Attack on **P-192** is made possible by our highly optimized parallel implementation.
- Attack on `sect163r1` is even feasible with a laptop.
- Recovering remaining bits is much cheaper in Bleichenbacher's framework.

ECDSA key recovery attack: experimental records

Target	Bias	Facility	Error rate	Input	Thread (Collision)	Time (Collision)	RAM (Collision)	Recovered MSBs
NIST P-192	1-bit	AWS EC2	0	2^{29}	96×24	113h	492GB	39
NIST P-192	1-bit	AWS EC2	1%	2^{35}	96×24	52h	492GB	39
`sect163r1`	1-bit	Cluster	0	2^{23}	16×16	7h	80GB	36
`sect163r1`	1-bit	Workstation	2.7%	2^{24}	48	42h	250GB	35
`sect163r1`	2-bit	Cluster	0	1024	16	2h	96GB	32

Table 1: Computational results for the first round of Bleichenbacher

- Attack on **P-192** is made possible by our highly optimized parallel implementation.
- Attack on `sect163r1` is even feasible with a laptop.
- Recovering remaining bits is much cheaper in Bleichenbacher's framework.

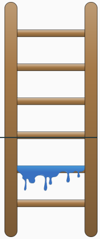
How to Acquire Nonce Leakage

LadderLeak: Tiny timing leakage from the Montgomery ladder

Algorithm 1 Montgomery ladder

Input: $P = (x, y)$, $k = (1, k_{t-2}, \ldots, k_1, k_0)$
Output: $Q = [k]P$

$k' \leftarrow$ Select $(k + q, k + 2q)$
$R_0 \leftarrow P$, $R_1 \leftarrow [2]P$
for $i \leftarrow \lg(q) - 1$ **downto** 0 **do**
    Swap (R_0, R_1) if $k'_i = 0$
    $R_0 \leftarrow R_0 \oplus R_1$; $R_1 \leftarrow 2R_1$
    Swap (R_0, R_1) if $k'_i = 0$
end for
return $Q = R_0$



Conditions for the attack to work:

- Accumulators (R_0, R_1) are in **projective coordinates**, but initialized with the base point in **affine coordinates**.
- Group order is $2^n - \delta$
- Group law is non-constant time wrt handling Z coordinates $\leadsto$ **Weierstrass model**

Experiments were carried out with Flush+Reload cache attack technique

$\leadsto$ MSB of k was detected with $> 99\ \%$ accuracy.

Software countermeasures & coordinated disclosure

- **Coordinated disclosure**: reported in December 2019 (before EOL of OpenSSL `1.0.2`)
- Fixed in April 2020 with **randomized** Z coordinates of the base point

Main takeaways

- ECDSA nonce is extremely sensitive!
  - Even <1-bit leakage/signature is exploitable, albeit with quite a few signatures as input
- HNP is still relevant nowadays
- Interesting connection between the HNP and GBP
  - Open question: Could #signatures for Bleichenbacher be as low as lattice?

Thank you! & Questions?

More details at `https://ia.cr/2020/615`

Main takeaways

- ECDSA nonce is extremely sensitive!
  - Even < 1-bit leakage/signature is exploitable, albeit with quite a few signatures as input
- HNP is still relevant nowadays
- Interesting connection between the HNP and GBP
  - Open question: Could #signatures for Bleichenbacher be as low as lattice?

Thank you! & Questions?

More details at `https://ia.cr/2020/615`

Main takeaways

- ECDSA nonce is extremely sensitive!
  - Even < 1-bit leakage/signature is exploitable, albeit with quite a few signatures as input
- HNP is still relevant nowadays
- Interesting connection between the HNP and GBP
  - Open question: Could #signatures for Bleichenbacher be as low as lattice?

Thank you! & Questions?

More details at `https://ia.cr/2020/615`

Main takeaways

- ECDSA nonce is extremely sensitive!
  - Even <1-bit leakage/signature is exploitable, albeit with quite a few signatures as input
- HNP is still relevant nowadays
- Interesting connection between the HNP and GBP
  - Open question: Could #signatures for Bleichenbacher be as low as lattice?

Thank you! & Questions?

More details at `https://ia.cr/2020/615`

References i

Daniel Bleichenbacher.
On the generation of one-time keys in DL signature schemes.
Presentation at IEEE P1363 working group meeting, 2000.

Dan Boneh and Ramarathnam Venkatesan.
Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes.
In Neal Koblitz, editor, *CRYPTO'96*, volume 1109 of *LNCS*, pages 129–142. Springer, Heidelberg, August 1996.

Alejandro Cabrera Aldaya, Billy Bob Brumley, Sohaib ul Hassan, Cesar Pereida García, and Nicola Tuveri.
Port contention for fun and profit.
In *2019 IEEE Symposium on Security and Privacy*, pages 870–887. IEEE Computer Society Press, May 2019.

Fergus Dall, Gabrielle De Micheli, Thomas Eisenbarth, Daniel Genkin, Nadia Heninger, Ahmad Moghimi, and Yuval Yarom.
CacheQuote: Efficiently recovering long-term secrets of SGX EPID via cache attacks.
IACR TCHES, 2018(2):171–191, 2018.
https://tches.iacr.org/index.php/TCHES/article/view/879.

Freepik.
Icons made by Freepik from Flaticon.com.
http://www.flaticon.com.

Nick Howgrave-Graham and Nigel Smart.
Lattice attacks on digital signature schemes.
Designs, Codes and Cryptography, 23(3):283–290, 2001.

Jan Jancar, Vladimir Sedlacek, Petr Svenda, and Marek Sýs.
Minerva: The curse of ECDSA nonces systematic analysis of lattice attacks on noisy leakage of bit-length of ECDSA nonces.
IACR Trans. Cryptogr. Hardw. Embed. Syst., 2020(4):281–308, 2020.

- Robert Merget, Marcus Brinkmann, Nimrod Aviram, Juraj Somorovsky, Johannes Mittmann, and Jörg Schwenk.
  Raccoon attack: Finding and exploiting most-significant-bit-oracles in tls-dh(e).
  Cryptology ePrint Archive, Report 2020/1151, 2020.
  https://eprint.iacr.org/2020/1151.
- Daniel Moghimi, Berk Sunar, Thomas Eisenbarth, and Nadia Heninger.
  TPM-FAIL: TPM meets timing and lattice attacks.
  CoRR, abs/1911.05673, 2019.
  To appear at USENIX Security 2020.

References v

Keegan Ryan.
Return of the hidden number problem.
IACR TCHES, 2019(1):146–168, 2018.
https://tches.iacr.org/index.php/TCHES/article/view/7337.

Sohaib ul Hassan, Iaroslav Gridin, Ignacio M. Delgado-Lozano, Cesar Pereida García, Jesús-Javier Chi-Domínguez, Alejandro Cabrera Aldaya, and Billy Bob Brumley.
Déjà vu: Side-channel analysis of mozilla's NSS.
In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, pages 1887–1902. ACM, 2020.