Centrum Wiskunde & Informatica

Cryptanalysis of the cryptographic standard SHA-1

Marc Stevens
Cryptology Group
CWI Amsterdam

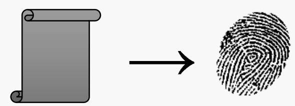
joint work with Pierre Karpman (Inria, NTU)
& Thomas Peyrin (NTU)

Background

Background

- **Cryptographic hash functions**

$$H: \{0,1\}^* \longrightarrow \{0,1\}^N$$



- **Collision resistance (informal)**

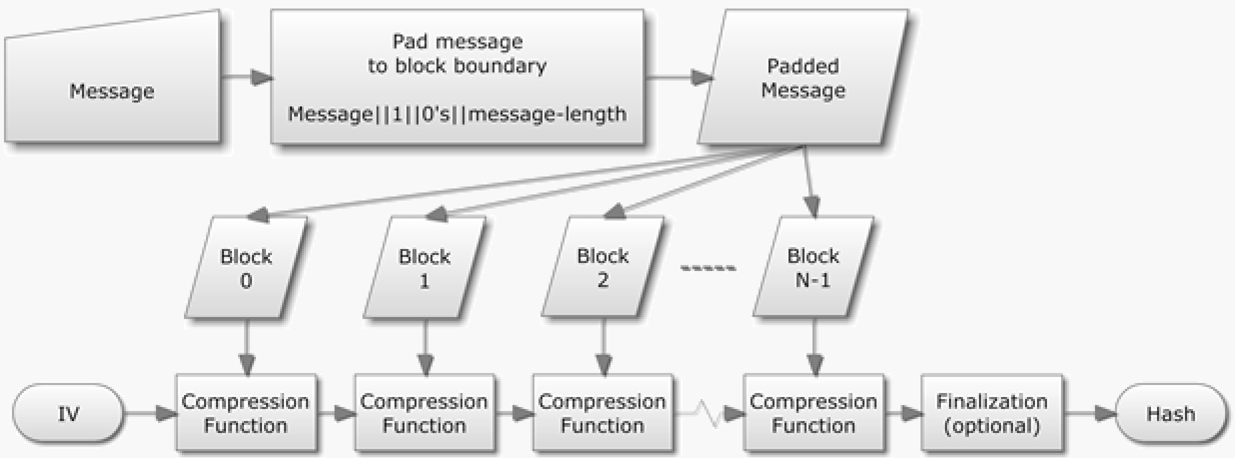
  Infeasible to find $x \neq y$ with SHA-1(x)=SHA-1(y)

  (Generic attack: $O(2^{N/2})$)

- Weak: MD5 [Riv92], SHA-1 [NIST1995]
- Secure: SHA-2 [NIST2001], SHA-3 [NIST2015]

Background

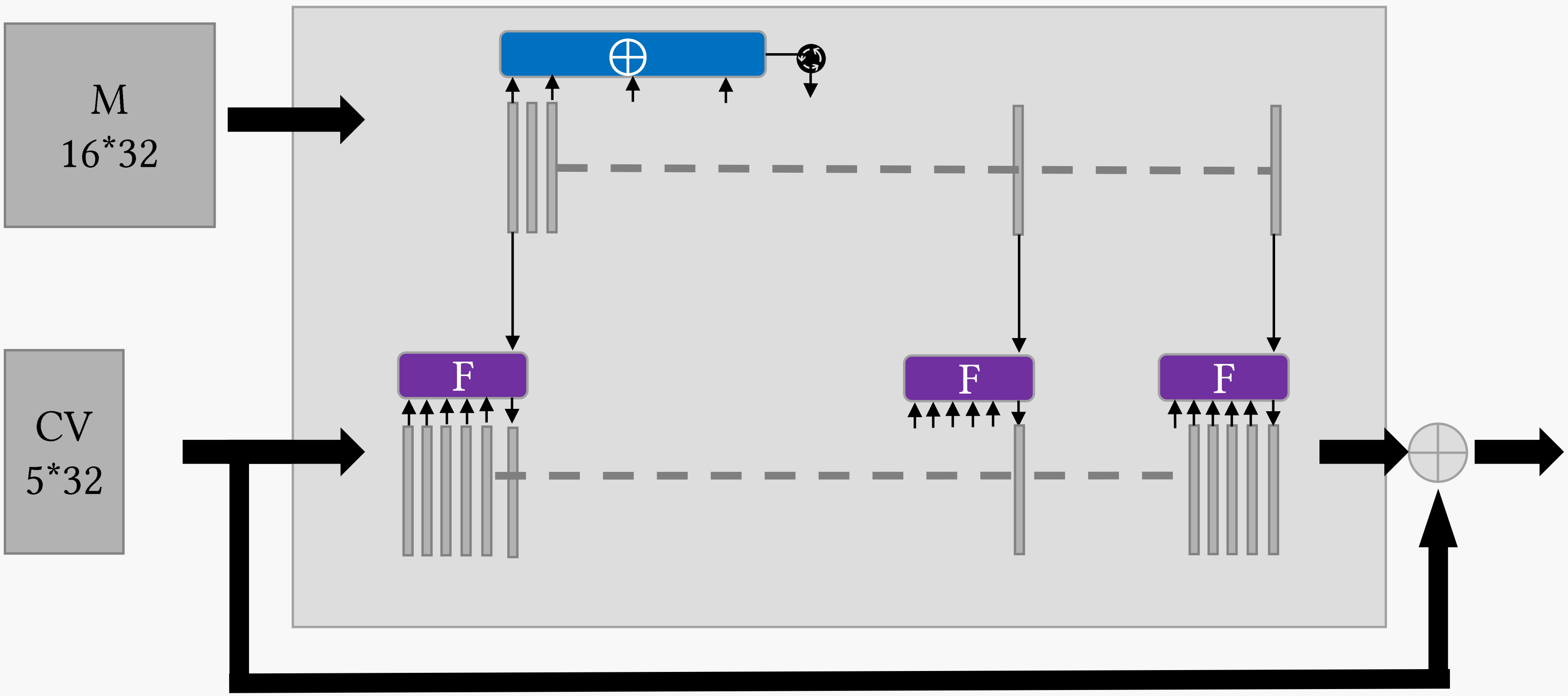
- **Merkle-Damgård Construction**
  - Splits message into 512-bit blocks
  - Processes them iteratively using compression function



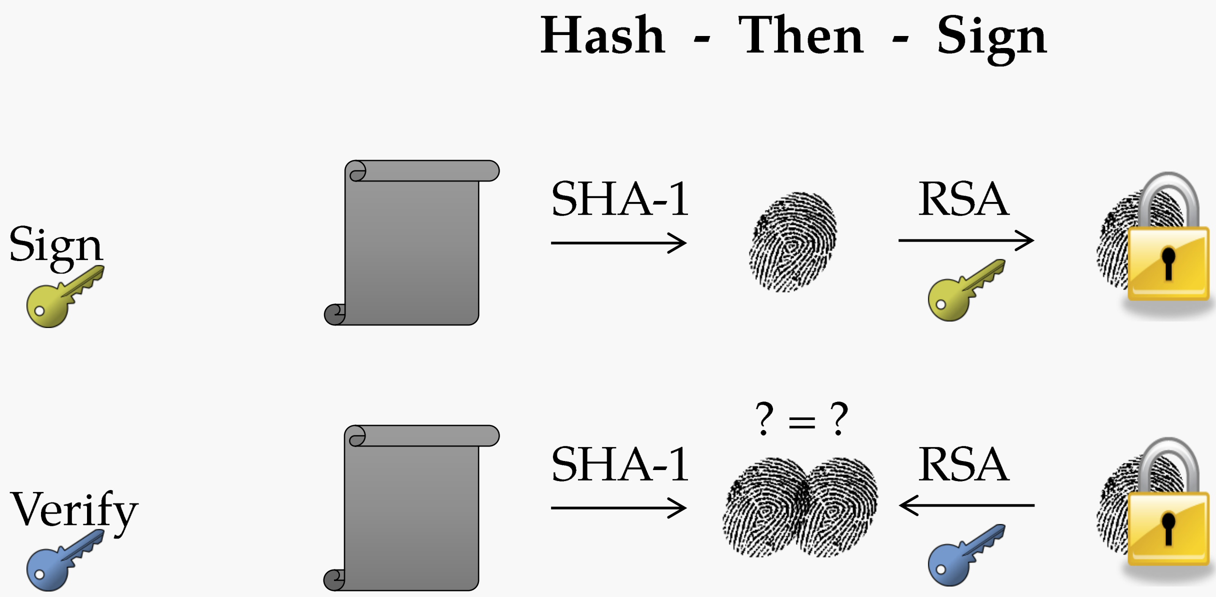
- Security proof
  - (H collision ⇒ C.F. collision)
  - ⇒ (C.F. collision resistant ⇒ H collision resistant)
  - ⇒ (C.F. collision ⇒ ? (no security proof))

SHA-1 Compression function

- Linearly expand 16 words (32-bits) of message to 80 words
- Non-linear step function on 5 state words & 1 message word
- Davies-Meyer feedforward of Chaining Value

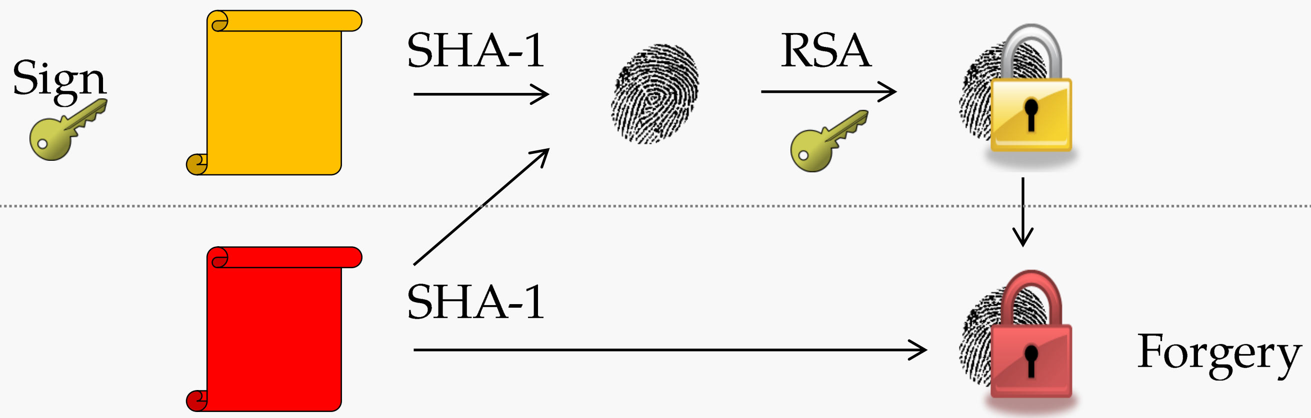


Digital signature standards based on



Widely-used standards: (MD5-RSA,) SHA-1-RSA, SHA-2-RSA

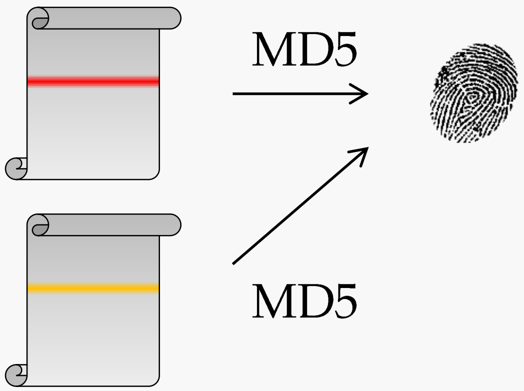
Security depends on **collision resistance** of hash function



Background

[Wang et al. 2004]

- Breakthrough cryptanalytic attacks

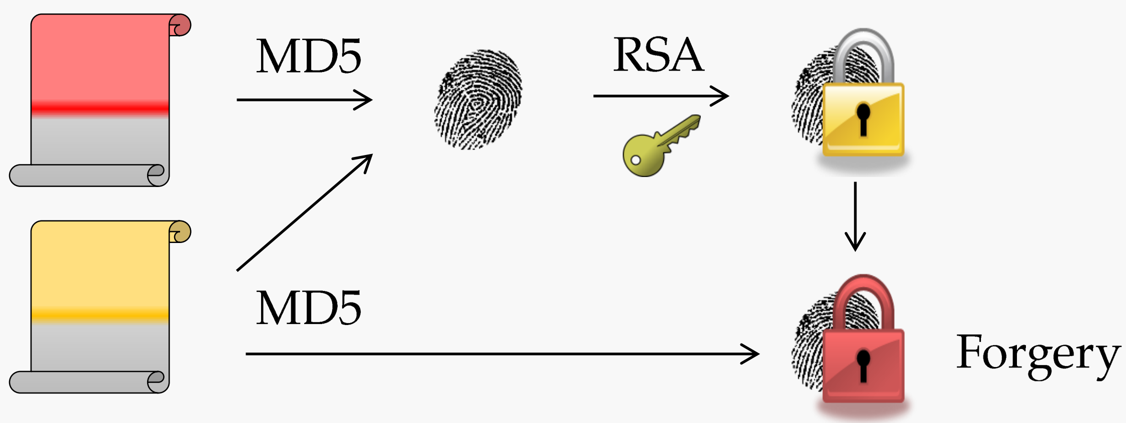


- Theoretical & practical break of hash function MD5
- Limited impact: **identical-prefix collisions**

Background

[2007&2009 Stevens et al.]

- more versatile: **chosen-prefix collision** attacks



- Practical: realistic abuse scenario with high impact

	MD5		SHA-1		SHA-256	
	Id.Pr.	Ch.Pr.	Id.Pr.	Ch.Pr.	Id.Pr.	Ch.Pr.
Birthday	$2^{64.3}$	$2^{64.8}$	$2^{80.3}$	$2^{80.8}$	$2^{128.3}$	$2^{128.8}$
2004	2^{40}		2^{69}			
2005	2^{37}		(2^{63})			
2006	2^{32}	2^{49}				
2007	2^{25}	2^{42}	(2^{61})			
2008	2^{21}					
2009	2^{16}	2^{39}				
2010						
2011						
2012			2^{61}	2^{77}		
today	$\mathbf{2^{16}}$	$\mathbf{2^{39}}$	$\mathbf{2^{61}}$	$\mathbf{2^{77}}$	$\mathbf{2^{128.3}}$	$\mathbf{2^{128.8}}$

Published collision attacks on MD5 & SHA-1

Background

- [NIST2011] Special Publication 800-131A

Hash Function	Use	
SHA-1	Digital signature generation	Acceptable: -2010 Deprecated: 2011-2013 Disallowed: 2014-
	Digital signature verification	Acceptable: -2010 Legacy-use: 2011-
	Other applications	Acceptable

IGNORED

- [Schneier2012]: Projected costs of SHA-1 collisions

$2.77M in 2012
$700K by 2015
$173K by 2018
$43K by 2021
(based on [Stevens12], Amazon EC2 rates & Moore's Law)

- Actual CA/Browser Forum policy:
  - SHA-1 digital signature generation up to 1 Jan. 2016 ~~(proposal: 1 Jan. 2017)~~
  - SHA-1 digital signature verification up to 1 Jan. 2017

Prior work

- (Identical-prefix) collision attacks on full SHA-1
  - Birthday search : $\mathbf{2^{80}}$
  - [WYY05] : $\mathbf{2^{69}}$
  - Wang, Yao, Yao 2005 : $\mathbf{2^{63}}$ (no publication, partially verified)
  - [SKI06] : **??** ($\mathbf{2^{52}}$ symbolic message modifications x2^{23}?)
  - Mendel et al. 2007 : $\mathbf{2^{60.x}}$ (no publication)
  - [MHP09] : $\mathbf{2^{52}}$ (withdrawn)
  - Chen 2011 : $\mathbf{2^{58}}$ (not peer-reviewed, too optimistic by factor $2^{3.5}$)
  - [Stevens13] : $\mathbf{2^{61}}$

- Example reduced-round SHA-1 collisions
  - [DR06] : $\mathbf{2^{35}}$ (64 out of 80 steps)
  - [DMR07] : $\mathbf{2^{44}}$ (70 out of 80 steps)
  - [Gre10] : $\mathbf{2^{50.7}}$ (73 out of 80 steps)
  - [GA11] : $\mathbf{2^{57.7}}$ (75 out of 80 steps) (10,000 GPU-days, 1GPU≈40cores)

Our work

Our work

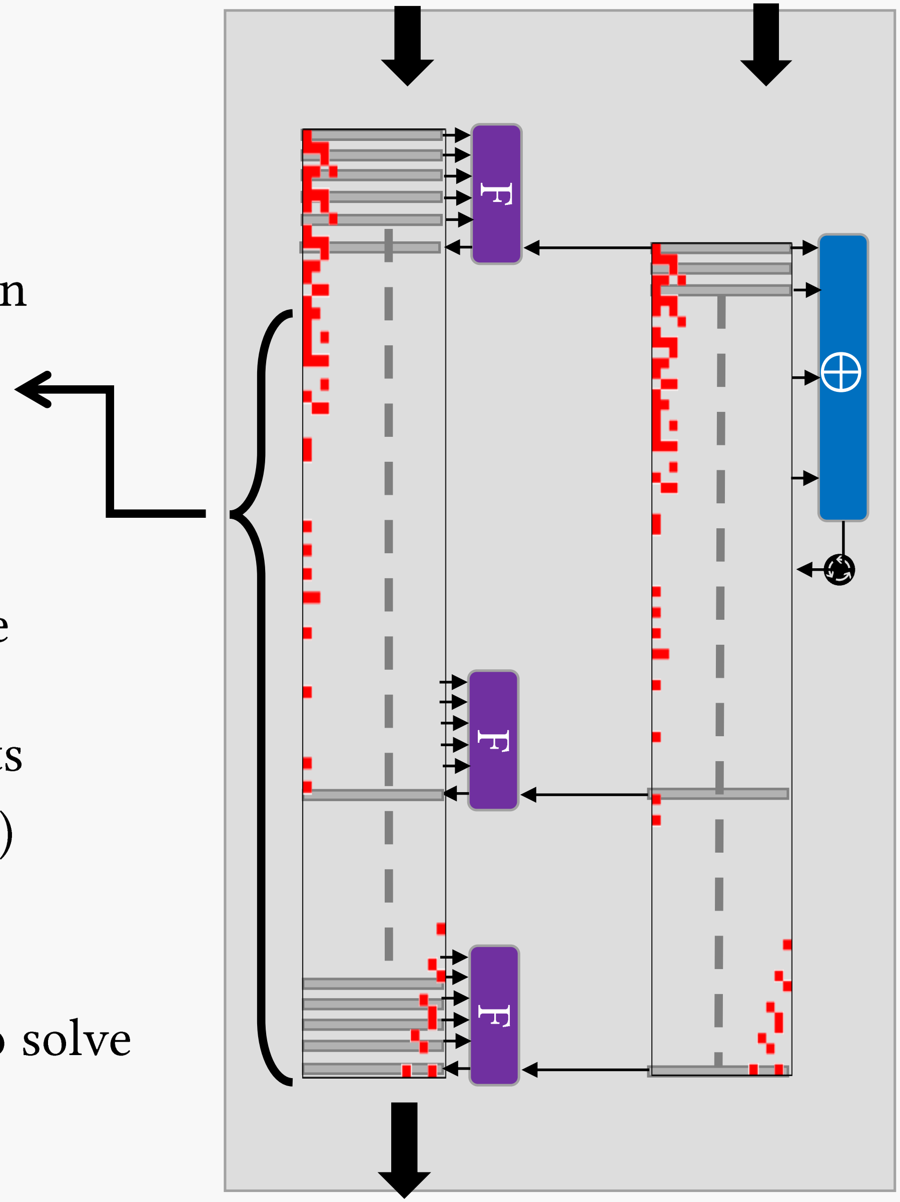
- Example SHA-1 collisions thought to be imminent since 2005
- Previous works show analysis more complicated & too high cost
- Our research directions
  1. Precise analysis
     ⇒ optimal complexity & degrees of freedom
  2. Use massively-parallel architectures: graphic cards (GPUs)
     ⇒ more cost efficient
  3. Collisions on (reduced-round) SHA-1's Compression Function
     ≡ freestart collision attack on (reduced-round) SHA-1
- Our results: freestart collision attacks on SHA-1
  - [KPS15] : $\mathbf{2^{50.3}}$ (76 out of 80 steps) (5 GPU-days, 1GPU≈140cores)
  - [SKP16] : $\mathbf{2^{57}}$ (**80 out of 80 steps**) (640 GPU-days, 1GPU≈140cores)
  - First practical attack on full SHA-1 !
  - More efficient GPU implementation (prev: 1GPU≈40cores)
  - Estimations for cost of collision attack on full SHA-1
    : $\mathbf{2^{61}}$ (SHA-1 collision) (40,000 GPU-days, EC2 ≈ **$100k**)

Differential path

- Precise description of how differences propagate through compression function
- Last 60 steps determine most of attack's complexity
- [Stevens13] precise methods to determine optimal differential paths

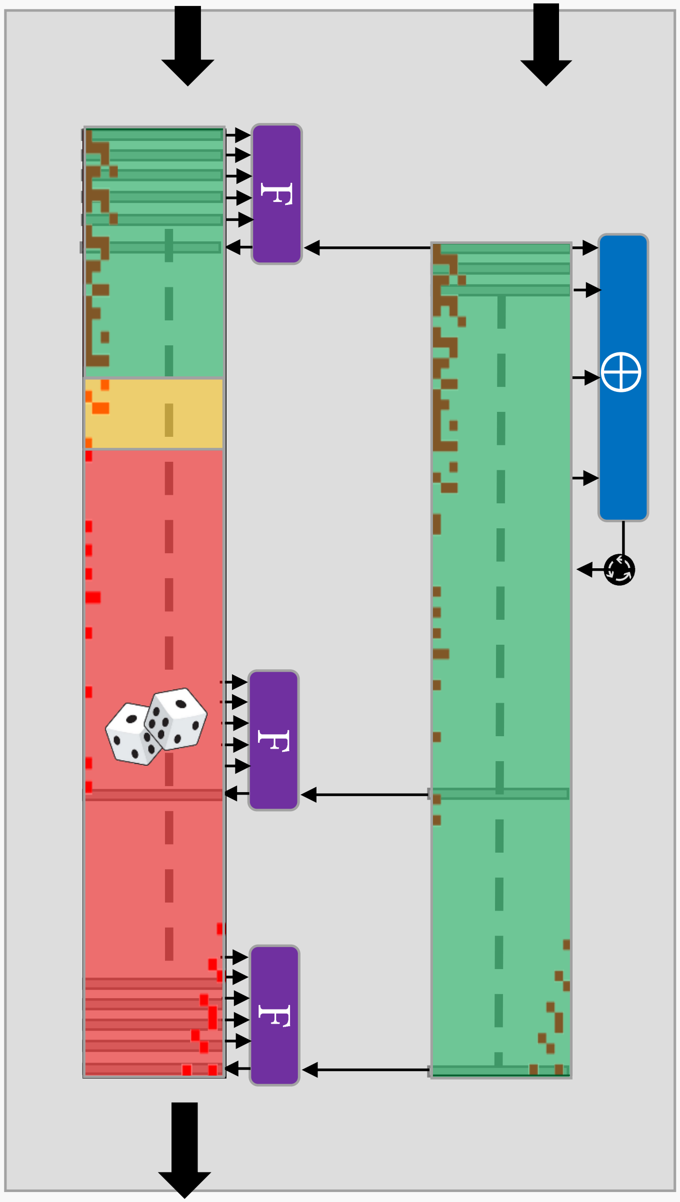
  [KPS15,SKP16] improvements

  (very technical, omitted here)
- Translate differential path into **system of equations** to solve



System of equations

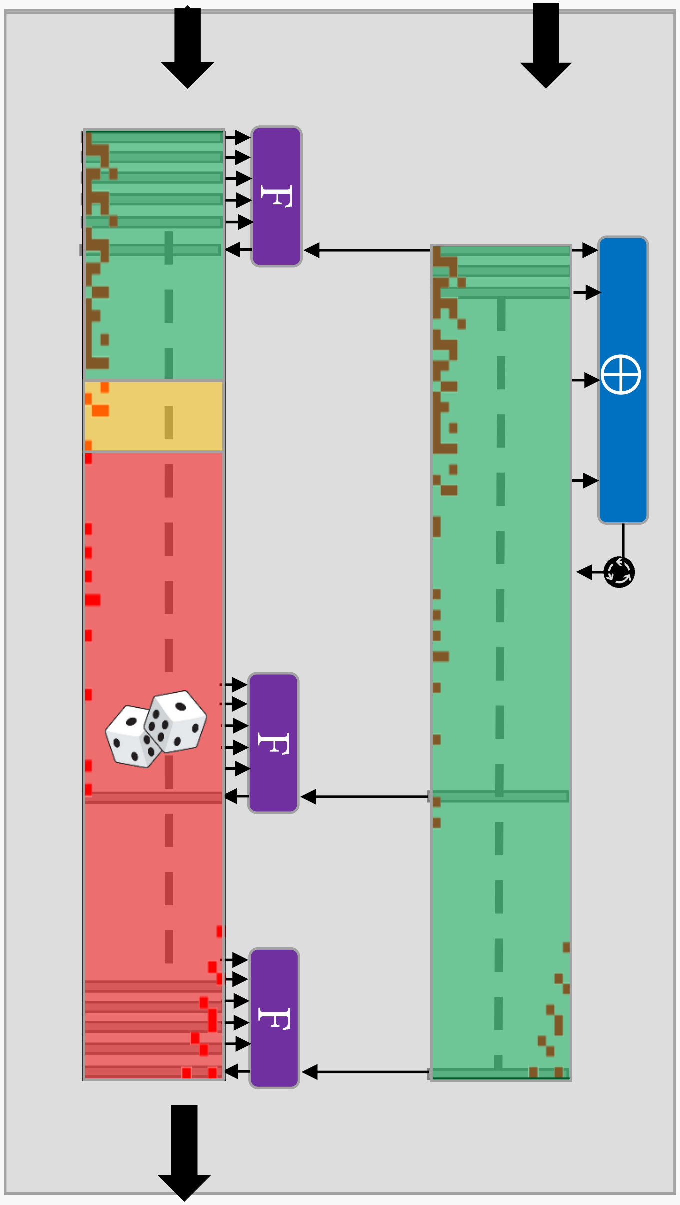
- Simple equations
  on expanded message bits
  ⇒ linear equations
  on input message bits
- Simple equations on state bits
- First 16 steps easily solved
  ⇒ all message bit equations fulfilled
  ⇒ determines remaining 64 steps
- Make predictable small changes
  to solve up to step 24
  (amortizes cost of earlier steps)
  ⇒ only control about 30% of SHA-1
- Find many solutions up to step 24
  to probabilistically fulfill
  remaining steps



Freestart collision attack

- Start from the middle
  - Advantage:
    higher probability diff.path
    ⇒ lower complexity
  - Disadvantage:
    cannot control input CV
  - ⇒ collision for C.F.

- Motivation
  - Invalidates security proof
  - Intermediate results
  - To perfect cryptanalysis tools
  - Testbed for GPU implementation



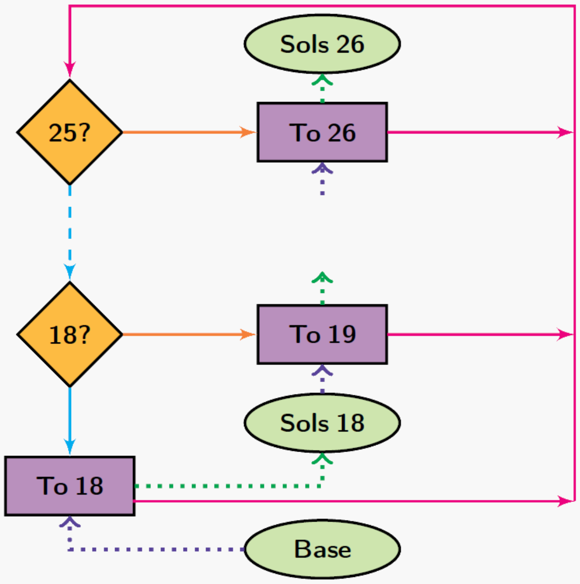
GPU

- Nvidia GTX-970
- Recent, high-end, good price/performance
- 13 x 128 = 1664 cores @ 1.2 GHz
- High-level programming with CUDA
- Throughput for 32-bit arithmetic: all 1/cycle/core (except rotl/rotr)
- ≈ € 350

- Single Instruction Multiple Threads
  - Execution is bundled in **warps** of 32 threads
  - Control-flow divergence is serialized ⇒ minimize branching
- Hide latency by running more threads than cores
  - Transparent scheduling of actionable warps to cores
- Be careful: incoherent memory reads/writes are slow

GPU

- [KPS15,SKP16] GPU tree search framework
  1. Store partial solutions up to some step in shared buffers
  2. Every thread of a warp loads one solution
  3. ... tries all degrees of freedom for this step
  4. ... stores successful larger partial solution in next step buffer
- Depth-first search: always process last queue with enough work



GPU vs CPU

- Freestart 76-step SHA-1 [KPS15]
  - Initial GPU implementation: only easy speed-up tricks
  - On one GPU, the attack takes ≈ 4.2 days
  - On one CPU core @ 3.2 GHz, the attack takes ≈ 606 days
  - ⇒ One GPU ≡ 140 CPUcores
  - (To compare with ≡ 40 [GA11])
  - For raw SHA-1 computations, ratio is 320
  - ⇒ Relative loss of only ×2.3 due to branching
    (better than expected for a highly branching tree search!)

- Freestart full SHA-1 (80-steps) [SKP16]
  - Second generation implementation: also advanced speed-up tricks
  - Complexity: 2^{57}
  - ≈10 days on 64 GPUs (16 desktops with 4 GTX970 each)
  - First practical attack on full SHA-1

	Message 1
IV_1	50 6b 01 78 ff 6d 18 90 20 22 91 fd 3a de 38 71 b2 c6 65 ea
M_1	9d 44 38 28 a5 ea 3d f0 86 ea a0 fa 77 83 a7 36
	33 24 48 4d af 70 2a aa a3 da b6 79 d8 a6 9e 2d
	54 38 20 ed a7 ff fb 52 d3 ff 49 3f c3 ff 55 1e
	fb ff d9 7f 55 fe ee f2 08 5a f3 12 08 86 88 a9
Compr(IV_1, M_1)	f0 20 48 6f 07 1b f1 10 53 54 7a 86 f4 a7 15 3b 3c 95 0f 4b

	Message 2
IV_2	50 6b 01 78 ff 6d 18 91 a0 22 91 fd 3a de 38 71 b2 c6 65 ea
M_2	3f 44 38 38 81 ea 3d ec a0 ea a0 ee 51 83 a7 2c
	33 24 48 5d ab 70 2a b6 6f da b6 6d d4 a6 9e 2f
	94 38 20 fd 13 ff fb 4e ef ff 49 3b 7f ff 55 04
	db ff d9 6f 71 fe ee ee e4 5a f3 06 04 86 88 ab
Compr(IV_2, M_2)	f0 20 48 6f 07 1b f1 10 53 54 7a 86 f4 a7 15 3b 3c 95 0f 4b

Full SHA-1

- Predictions for cost of collisions for full SHA-1
  - Complexity: 2^{61} [Stevens13]
  - $\approx$ 40,000 GPU days (Amazon EC2: older GPUs)
  - $\approx$ $100k renting fee on Amazon EC2 (spot-prices)
  - $\times$ 7 lower cost in 2015 than predicted earlier by Schneier

Impact & Conclusion

Industry Impact

- CA/Browser Forum: Ballot 152
  - Extend issuance SHA-1 certificates up to 1 Jan. 2017 (before: 1 Jan. 2016)
  - (unaltered: deprecate SHA-1 certificates after 1 Jan. 2017)
  - Proposed/endorsed by Entrust, Microsoft, Trend Micro
  - Seemingly enough support to pass
  - Our recommendations on 8 Oct. ensured Ballot did not pass on 16 Oct.

- Certification Authorities have found loop-hole
  - Withdraw older CA certificate from Browser root-CA-stores
  - ⇒ not encumbered by CA/Browser regulations ⇒ can sign SHA-1

- Mozilla, Microsoft & Google:
  - Possibly deprecate SHA-1 certificates per 1 July 2016

- TLS 1.3 draft 9
  - Deprecated all uses of SHA-1 digital signatures

Conclusion

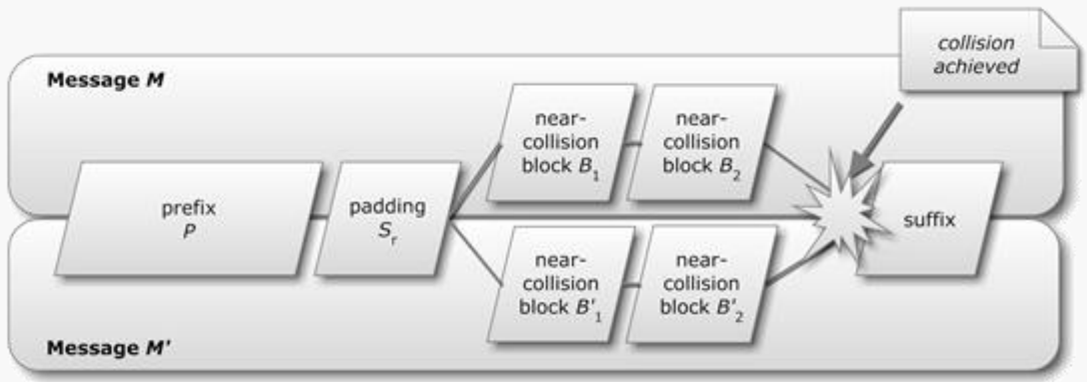
- Improved cryptanalysis of SHA-1 using
  - Precise analysis methods (omitted here)
  - More efficient GPU tree search framework
- Freestart collision attacks on
  - 76-step SHA-1
  - Full SHA-1 ! ⇒ first practical attack on full SHA-1
  - ⇒ invalidates SHA-1's collision resistance proof
- Work-in-progress
  - Collision attack on full SHA-1
- Industry is deprecating SHA-1 painstakingly slow
  - SHA-1 has been used ubiquitously as de facto industry standard
    ⇒ very hard and costly to deprecate everywhere
  - CA/Browser forum is at the frontier, but deprecating per 1 Jan. 2017
  - ⇒ Need practical examples to speed-up deprecation
- Note: counter-cryptanalysis [Stevens13b]
  - Detect digital signature forgeries constructed using collision attack
  - Practical & real-time: only $\times$ 2 as slow as plain SHA-1

Thank you!

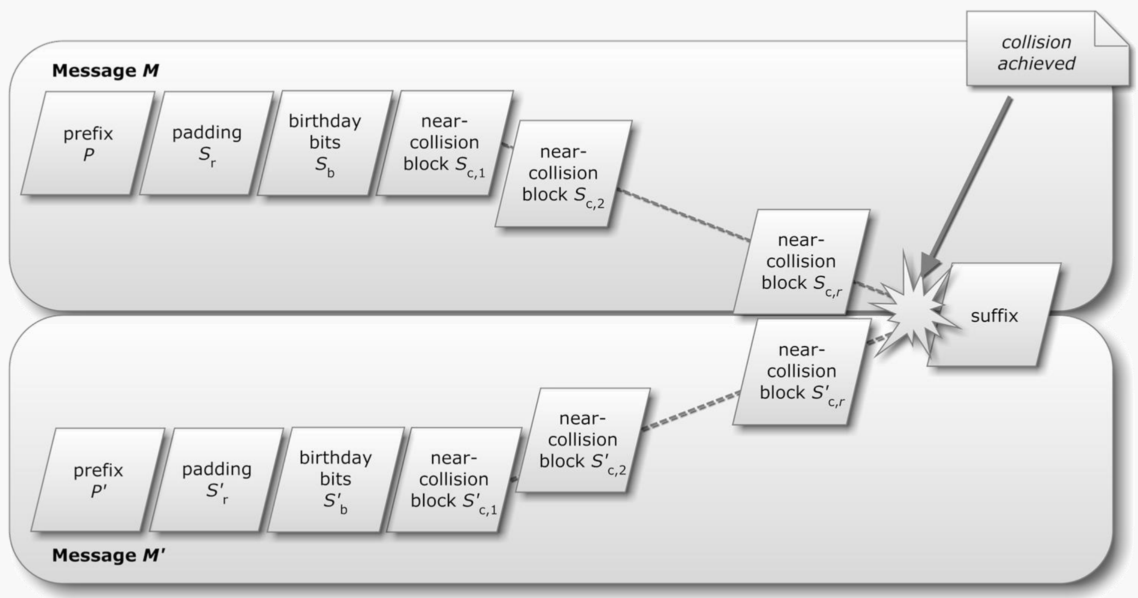
SHA-1 cryptanalysis

Attacks on SHA-1 based on near-collision attacks

Identical-prefix collision attack



Chosen-prefix collision attack



SHA-1 cryptanalysis

- Attacks on SHA-1 based on near-collision attacks
- Near-collision attack on compression function:
  - Given input chaining value pair
  - Compute message block pair
  - To achieve ‘desired’ difference between output chaining values