Improving Counter-cryptanalysis

Marc Stevens
marc.stevens @ cwi.nl

CWI Amsterdam

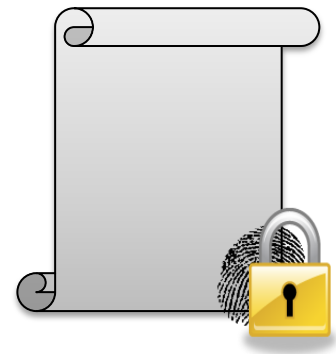
Part I – Weak signature schemes

Part II – Counter-cryptanalysis
Part III – Flame
Part IV – Improvements
To Conclude…

Digital signature schemes

- One of the pillars for P.K.I.s

- Used to ensure authenticity in/of
  - Browsers
  - Documents
  - Email
  - Software updates
  - Downloadable content
  - Currency transactions



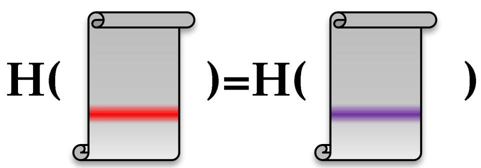
- Hash-Then-Sign:
  {MD5,SHA-1,SHA-2}-{RSA,DSA}

- Hash collision MD5(A)=MD5(B) $\Rightarrow$ forgery

Collision attacks on MD5 & SHA-1

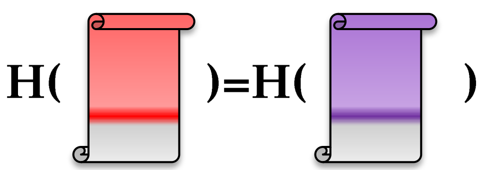
- Distinguish between 2 types
  - Identical prefix

    H(P|C|S)=H(P|C′|S)

  - Chosen-prefix

    H(P|C|S)=H(P′|C′|S)

  - P, P′, S: Free to choose s/t |P|=|P′|
  - C, C′: Generated based on P and P′, |C|=|C′|∈ [64B,1KB]

	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

Notes

- Generate your own MD5 chosen-prefix collision attack in a day using Project HashClash: https://code.google.com/p/hashclash/
- No publicly known collision for SHA-1 has been found yet
- First SHA-1 collision more likely to be constructed by nation-states than academia due to required resources, see: http://www.schneier.com/blog/archives/2012/10/when_will_we_se.html

Strategies for *meaningful* colliding files

- Using identical-prefix collisions
  - Meaningful C and C′ — Hard

    C = "… of money is $10,000.00…"
    C′ = "… of money is $20,000.00 …"

    C = "… OFFSET=X …"
    C′ = "… OFFSET=Y …"

  - IF-THEN-ELSE construct — Easy, but requires IF-THEN-ELSE

    IF (C ==C) THEN … ELSE …
    IF (C′==C) THEN … ELSE …

- Using chosen-prefix collisions
  - Meaningful different P, P′ & hide C and C′ in message — Easy

    P = "I owe you $20"   C = <hidden image>
    P′ = "You will inherit all my possessions"   C′ = <hidden image>

- Identical-prefix
  - Colliding Software [Kam04,Mik04]
  - Colliding PostScript documents [DL05,GIS05]
  - Colliding X.509 certificates (same ID, diff. RSA moduli) [LdW05]
- Chosen-prefix
  - Colliding PDF documents [SLdW07]
  - Colliding Software [SLdW07]
  - Colliding X.509 certificates (diff. IDs) [SLdW07]
  - Rogue Certification Authority [SSALMOdW09]
  - Rogue Windows Update signing certificate [Flame12]

What to do when a signature scheme is broken?

The easy answer: “migrate to a more secure scheme”
i.e., move from MD5-RSA to SHA-2-RSA

Who should migrate?

Signers: generate SHA-2-RSA signatures
Problems: compatibility/deployment issues, risk-cost trade-off, human,…
Result: forgeries can still be constructed till the last signer migrates

Verifiers: don’t accept MD5-RSA signatures
Problem: too many old signatures in use to just invalidate them all at once
Result: old and new forgeries can abused against nearly everyone

The easy answer is not a practical solution for the near future

Our answer: detect forged signatures

Verifiers: don’t accept *forged* MD5-RSA signatures
Results:
- Old legitimate signatures are still valid
- Verifier protected against forgeries
- Independent of migration by signers

How to detect forged signatures?: counter-cryptanalysis!

Part II – Counter-cryptanalysis

Part III – Flame
Part IV – Improvements
To Conclude…

New paradigm: counter-cryptanalysis

- Strengthen weak cryptographic primitives
- Detect cryptanalytic attacks at the cryptographic level
- Counter-cryptanalysis in principle enables the continued secure use of weak cryptographic primitives
- No strengthened redesign $\Rightarrow$ no compatibility issues
- May be used during migration to strengthened redesigns in the real world

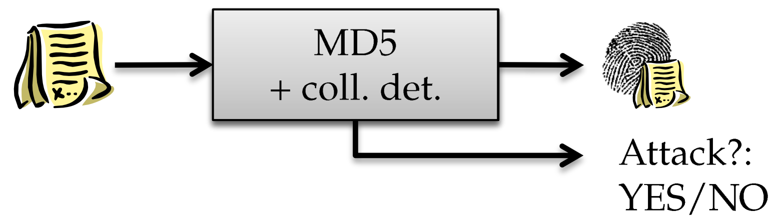
Why is that possible?

- Dedicated cryptanalytic attacks are highly specialized
- Active attacks may introduce subtle unavoidable anomalies
- Similar cryptanalytic techniques can be used to detect those anomalies
- This approach may detect an entire class of attacks that all introduce the same unavoidable anomalies

Counter-cryptanalysis
Collision detection

First practical example: collision detection

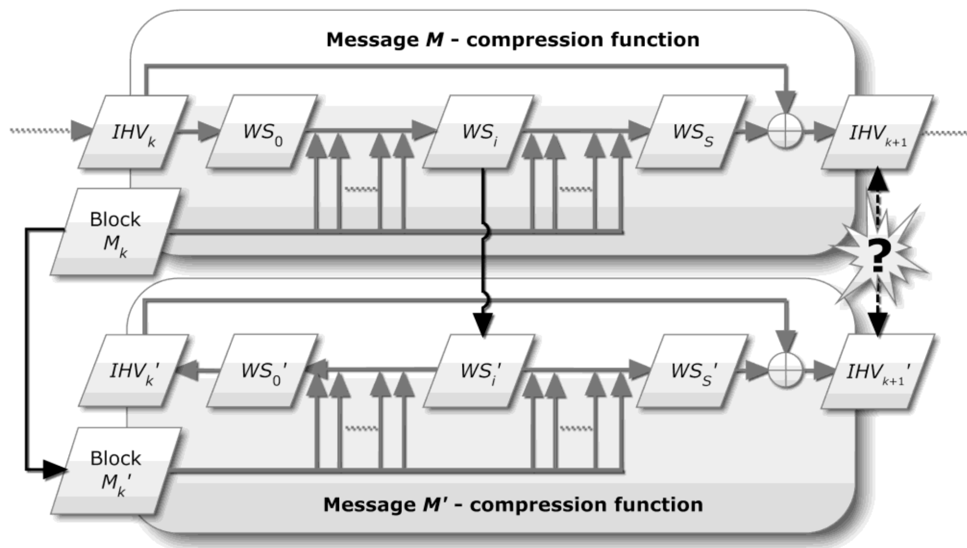
- Detect whether message was constructed using collision attack
- Single message of collision pair sufficient
- Application to MD5 & SHA-1
- Computational cost
  - MD5 factor x 224
  - SHA-1 factor x 15
  - Much less using early-abort: WIP



- Based on crucial properties of the known cryptanalysis on MD5 & SHA-1
  - Attacks exploit trivial differential steps with probability (close to) 1 to be able to obtain 'low' complexity
  - Very few message block differences result in attacks with 'low' complexity

Basic algorithm: detect last near-collision block

1. Guess message block difference & difference at trivial step i
2. Determine M_k' from M_k and WS_i' from WS_i
3. Reconstruct computation
4. Check whether collision in chaining value is obtained



If guess was correct then collision is detected with certainty

If guess was incorrect then a false positive occurs with probability $\approx 2^{-N}$

Reference implementation to detect collision attacks

- Available at http://marc-stevens.nl/research (at the bottom)

- Library interface to replace existing MD5/SHA-1 implementation
  - *MD5Init/MD5Init_unsafe, MD5Update, MD5Final*
  - *SHA1Init/SHA1Init_unsafe, SHA1Update, SHA1Final*
  - *{MD5,SHA1}Final* returns non-zero value if an attack is detected
  - *{MD5,SHA1}Init_unsafe* always results in correct (and possibly unsafe) hash
  - *{MD5,SHA1}Init* results in correct hash **if no attack has been detected**, otherwise a safe hash is returned

- Command line program
  - detectcollv <files>

Anomaly detection for digital signatures

- Online: active protection
  - Signer: protection against malicious signature requests
  - Verifier: protection against forged signatures
  - E.g., for TLS/SSL, OSs (drivers, executables, updates), etc.

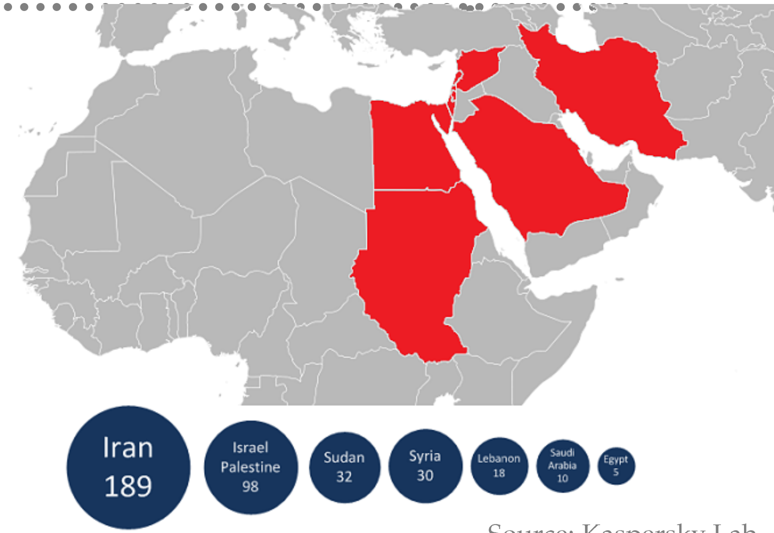
- Offline: forensic analysis
  - Main example: spyware Flame

Part III – Flame

Part IV – Improvements
To Conclude…

Cf. [Kas12,Sot12]

- Highly advanced malware
- Targeting the Middle-East
- Discovered in May 2012
- Active since 2007 or earlier
- Uncharacteristic features for malware
  - Up to 20 modules: each carefully selected prior to infection
  - Almost 20MB: includes Lua VM & libraries for compression, database, …
  - Did not spread wildly & evaded discovery for ~5 years
  - Surgical-precision attacks: each target carefully selected
  - Spread itself illegitimately using the Windows Update platform
  - First cryptanalytic attack on hash function found in the ‘wild’
  - Developed new variant cryptanalytic attack to do so…



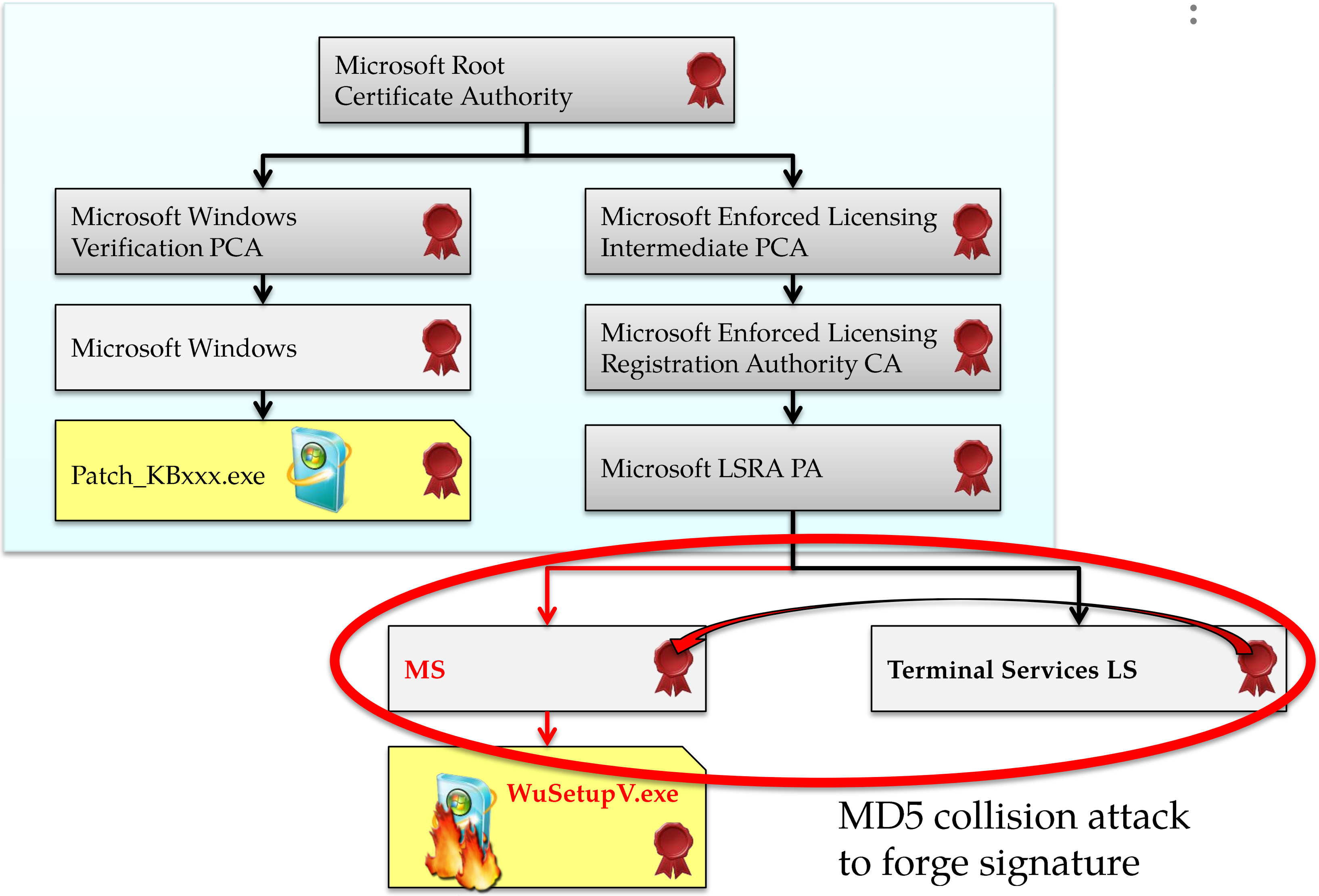
Source: Kaspersky Lab

Flame Propagation

- Man-in-the-middle attack on Windows Update
- Local network attack
  - Registers itself as proxy server for **update.microsoft.com** using WPAD (Web Proxy Auto-Discovery)
  - Windows Update falls back to insecure HTTP
    - depends on digital signatures for security
    - no need to subvert TLS/SSL connection

- Propagation
  - Flame serves fake 'security update' using Windows Update platform
  - Requires properly-signed 'security update'
  - **Uses illegitimate sub-CA** valid since **Feb 2010**
    ⇒ sub-CA invalid before that time
    ⇒ this attack was almost certainly done around Feb 2010 or later

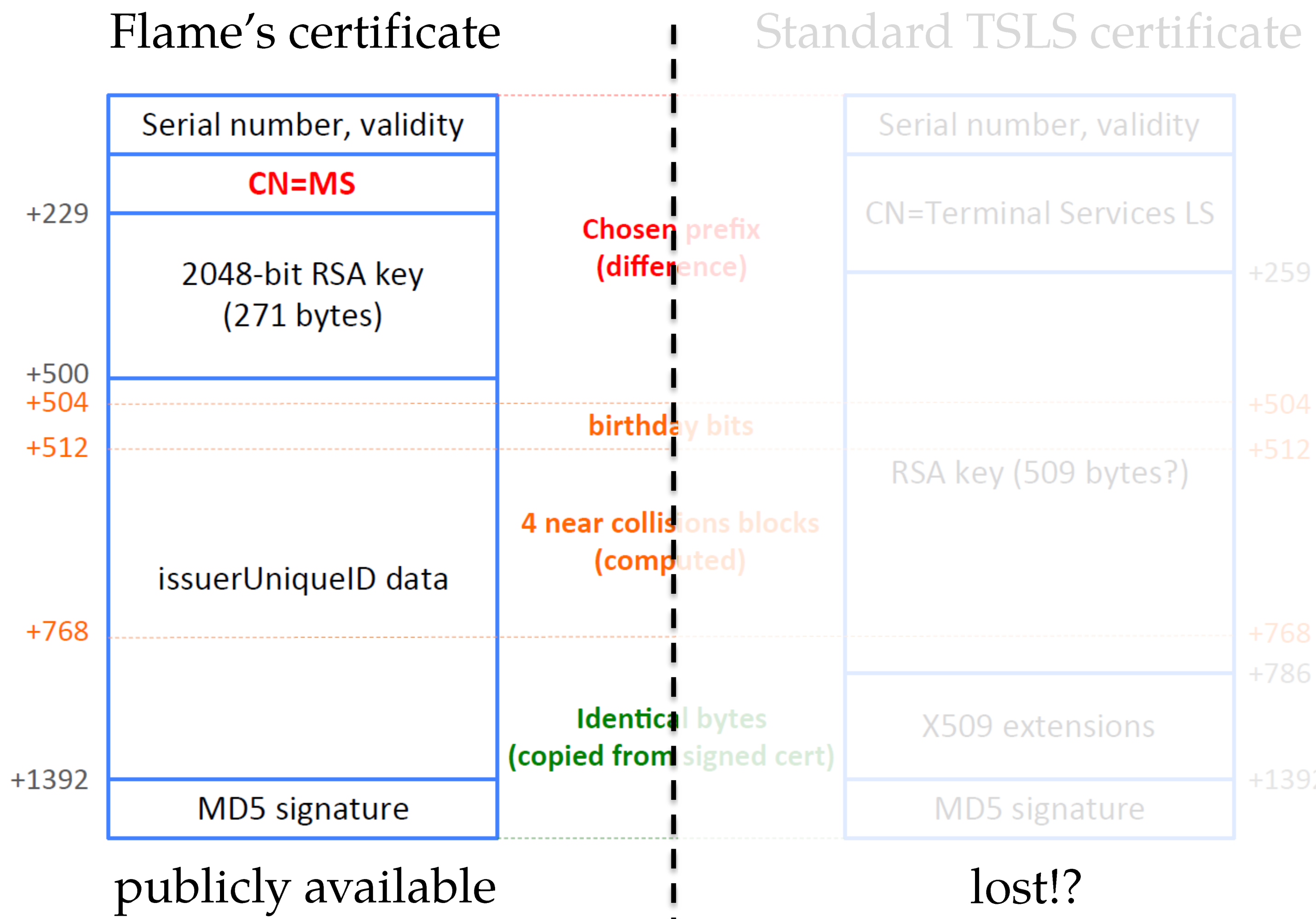
Flame
Certificate hierarchy



MD5 collision attack
to forge signature

Flame
Chosen-prefix collision attack

Uses chosen-prefix collision attack [SLdW07]:



Source: Alex Sotirov

Flame
Analyzing Flame's attack

- Only Flame's "MS" sub-CA certificate public
- The colliding "TSLS" certificate is not public (lost?)

- First example for counter-cryptanalysis
  - Assumed chosen-prefix collision attack
  - Only 1 of the 2 colliding certificates available to us
  - Ran proof-of-concept implementation (from 2008)
    - chosen-prefix collision detected
    - 4 near-collision blocks recovered
    - all differential paths reconstructed
    - <0.03 seconds
  - Differential paths expose use of new variant attack

Flame
Recovered differential paths

```
dm4=[!31!] dm11=[!15!] dm14=[!31!]
Q-3:            |........ ........ ........ ..-.....|
Q-2:            |00...... .1.1.01. ...1..+. ..-.10..|
Q-1:            |110-+..1 .1.-.00. .+.+.... ..-110..|
Q0:             |+-100..0 .-0+^++1 .0.+0.11 .110-+..| ok p=1
Q1:             |0+-++..- .-0++-+0 011-0..1 110+++..| ok p=0.49707
Q2:             |+0-0-.00 .-++00+- 0-1-+.1+ 1+-0++^.| ok p=0.166016
Q3:             |+010-000 .-+++0+1 +--.+^1+ -+-+++-.| ok p=1
Q4:             |-00-10+. .11-+-0+ +++11--0 -101-+0.| ok p=1
Q5:             |0-+-++-^ ^0110+1- -110+0-0 -0001+1^| ok p=1
Q6:             |++----+- ---+---- -----+++ ++++++++| ok p=1
Q7:             |111.-111 1101011. 110-1001 +0100.00| ok p=1
Q8:             |00+0.111 10111101 -1101100 .1110011| ok p=0.170898
Q9:             |..0.1... .....-.. 0.10+... 0-....0.| ok p=0.563477
Q10:            |..0^...1 ^....0.. 0^0-1... .1....+.| ok p=0.121094
Q11:            |..0-...1 +....-.. .+-01... .0..^.1.| ok p=0.899414
Q12:            |.1-1..^+ 1....+.. .0+0.... ....+.1.| ok p=0.946289
Q13:            |.0+1..-+ 1....0.. 100....1 ....0...| ok p=0.655273
Q14:            |.-+...1. .....1.. 1.+....1 ....1...| ok p=0.578125
Q15:            |.0+...10 ........ -.0....- ....-...| ok p=0.989258
Q16:            |.1+..... .0...... ..^..... ........| ok p=0.887695
Q17:            |..1..... .1....0. ^......^ ....^...| ok p=1
Q18:            |..0..... .+....1. ........ ........| ok p=0.998047
Q19:            |........ ......-. ........ ........| ok p=0.864258
Q20:            |0....... .^...... ........ ........| ok p=1
Q21:            |0....... ......^. ........ ........| ok p=0.501953
Q22:            |-....... ........ ........ ........| ok p=0.517578
Q23:            |........ ........ ........ ........| ok p=1
Q24:            |^....... ........ ........ ........| ok p=1
--------------------------------------------------------------------
Q25-32:         |........ ........ ........ ........| ok p=1
--------------------------------------------------------------------
Q33:            |0....... ........ ........ ........| ok p=1
Q34:            |1....... ........ ........ ........| ok p=0.507812
--------------------------------------------------------------------
Q35-59:         |±....... ........ ........ ........| ok p=1
--------------------------------------------------------------------
Q60:            |+.11110. ........ ........ ........| ok p=1
Q61:            |+.11000. ........ .001.00. ........| ok p=1
Q62:            |-.+----. ........ ...0.... ........| ok p=0.426758
Q63:            |+.?0??+. ........ .--+.+-. ........| ok p=0.855469
Q64:            |+......+ ++++++.. -..-.+-. .....+-.|
```

differential path 1

```
dm4=[!31!] dm11=[!-15!] dm14=[!31!]
Q-3:            |+....... ........ ........ ..-.....|
Q-2:            |-1....+. .1.1.0.. 0....1+. .-+...0.|
Q-1:            |+01.-.+1 .0-+.0^. 011+---1 -++.0.10|
Q0:             |1-0.1.+0 ^-0+1+-1 -1011+-0 001.1^-1| ok p=0.749023
Q1:             |10-.01.+ +++-0+10 --+111+- +--0-+1-| ok p=0.425781
Q2:             |.01.-011 00+-++0+ 0--+.--0 ++10+0+0| ok p=0.492188
Q3:             |..1.-+11 +001++^+ 01-+0110 0+1++0++| ok p=0.833008
Q4:             |..-.1-11 ++1-++-+ -1111--+ ++0+-+-1| ok p=1
Q5:             |^^1^+1-- 10-01011 0+10-1-+ 0-+++000| ok p=0.499023
Q6:             |+-++++++ ++++---- ------+- --+-----| ok p=1
Q7:             |0010-000 01111011 1011-111 10.10010| ok p=1
Q8:             |00000100 1111111+ -1001111 1-010111| ok p=0.672852
Q9:             |...-1... .-.....1 0..1+... .1....^.| ok p=0.495117
Q10:            |...0...0 ^0.....0 1..+0... .0....-.| ok p=0.895508
Q11:            |..0+..^0 -1...^.. ...01... ......1.| ok p=0.807617
Q12:            |.001..-+ 0....-.. ..01.... ......1.| ok p=1
Q13:            |.1-1..0- 1....0.. 1^1....1 ....1...| ok p=1
Q14:            |.-+...10 .....0.. 1-+....1 ....1...| ok p=0.586914
Q15:            |.0+....0 ........ +01....+ ....-...| ok p=0.994141
Q16:            |.^+..... .0...... .^^..... ........| ok p=0.879883
Q17:            |..1..... .1....0. ^......^ ....^...| ok p=1
Q18:            |..0..... .-....1. ........ ........| ok p=0.999023
Q19:            |........ ......-. ........ ........| ok p=0.895508
Q20:            |0....... .^...... ........ ........| ok p=1
Q21:            |0....... ......^. ........ ........| ok p=0.487305
Q22:            |-....... ........ ........ ........| ok p=0.508789
Q23:            |........ ........ ........ ........| ok p=1
Q24:            |^....... ........ ........ ........| ok p=1
--------------------------------------------------------------------
Q25-32:         |........ ........ ........ ........| ok p=1
--------------------------------------------------------------------
Q33:            |1....... ........ ........ ........| ok p=1
Q34:            |0....... ........ ........ ........| ok p=0.507812
--------------------------------------------------------------------
Q35-58:         |±....... ........ ........ ........| ok p=1
--------------------------------------------------------------------
Q59:            |+....... ........ ........ ..0.....| ok p=1
Q60:            |+.....0. ........ ...1001. 110.....| ok p=0.506836
Q61:            |-....100 ...0.... ...1..1. 00+.....| ok p=0.749023
Q62:            |+....1-. ........ ...-+++. +--.....| ok p=0.948242
Q63:            |+....++- ...+.... ...???-. ?+-.....| ok p=0.261719
Q64:            |......-- ..+..... .-....-. .+-....+|
```

differential path 2

Flame
Recovered differential paths

```
dm4=[!31!] dm11=[!15!] dm14=[!31!]
Q-3:           |........ ........ ........ ........|
Q-2:           |.1.10100 .....11. 10...... ..0.....|
Q-1:           |^0.0101- .1.0^10. 11.0.... ..1.100^|
Q0:            |++1-++++ 1001---. --.1.... .1+.110-| ok p=1
Q1:            |0-111110 1-1+1+-^ --1+.... .01^++-0| ok p=0.96875
Q2:            |10-01110 +++1---+ +10+.... 0-0++++1| ok p=0.374023
Q3:            |-0-01^1+ +0+1--10 0-++^^.0 01+0+00.| ok p=1
Q4:            |--0++-00 0-0+11++ ++-1-+10 -+00+-1.| ok p=1
Q5:            |-1++-0-1 +1-00+1- +0++110- -1--1+^^| ok p=1
Q6:            |++----+- ---+---- -----+++ ++++++++| ok p=1
Q7:            |1000-010 00.1010. 101-0101 +0001.00| ok p=1
Q8:            |11+1.101 01011100 -1000101 .1000011| ok p=0.0566406
Q9:            |..0.1... .....-.. 0.10+... 0-....0.| ok p=0.573242
Q10:           |..0^...1 ^....0.. 0^0-1... .1....+.| ok p=0.120117
Q11:           |..0-...1 +....-.. .+-01... .0..^.1.| ok p=0.889648
Q12:           |.1-1..^+ 1....+.. .0+0.... ....+.1.| ok p=0.948242
Q13:           |.0+1..-+ 1....0.. 100....1 ....0...| ok p=0.631836
Q14:           |.-+...1. .....1.. 1.+....1 ....1...| ok p=0.585938
Q15:           |.0+...10 ........ -.0....- ....-...| ok p=0.993164
Q16:           |.1+..... .0...... ..^..... ........| ok p=0.868164
Q17:           |..1..... .1....0. ^......^ ....^...| ok p=1
Q18:           |..0..... .+....1. ........ ........| ok p=0.999023
Q19:           |........ ......-. ........ ........| ok p=0.868164
Q20:           |0....... .^...... ........ ........| ok p=1
Q21:           |0....... ......^. ........ ........| ok p=0.495117
Q22:           |-....... ........ ........ ........| ok p=0.509766
Q23:           |........ ........ ........ ........| ok p=1
Q24:           |^....... ........ ........ ........| ok p=1
----------------------------------------------------------------
Q25-32:        |........ ........ ........ ........| ok p=1
----------------------------------------------------------------
Q33:           |1....... ........ ........ ........| ok p=1
Q34:           |1....... ........ ........ ........| ok p=0.493164
----------------------------------------------------------------
Q35-59:        |±....... ........ ........ ........| ok p=1
----------------------------------------------------------------
Q60:           |-.....0. ........ ......1. ........| ok p=1
Q61:           |-.0110.0 ........ .1....0. ........| ok p=0.514648
Q62:           |+..01.+. ........ .0....+. ........| ok p=0.492188
Q63:           |+.+---?- ........ .-....+. ........| ok p=0.395508
Q64:           |.+...+.- ....++++ -.....+. ....-...|
```

differential path 3

```
dm4=[!31!] dm11=[!-15!] dm14=[!31!]
Q-3:           |+....... ........ ........ ........|
Q-2:           |+....0+. ........ 000+---. ..000..1|
Q-1:           |+....+-. 11...-++ ++1101+. 10011..1|
Q0:            |001.1+-. 01^.^111 -++----0 11+-+11-| ok p=1
Q1:            |011.0.+. -+-^++1+ ++0000-1 +--0-11+| ok p=0.742188
Q2:            |+--.-0-. -+1+0--0 1+1-1-++ -1-00+--| ok p=0.756836
Q3:            |+--1-^1. .+100--+ 10---1+0 ---0++-1| ok p=1
Q4:            |-010+-1. 10-1-01+ 0-000-1- 0+-10-1-| ok p=1
Q5:            |+00-+00^ 0++-11-0 +++0-111 01-+-100| ok p=1
Q6:            |+-++++++ ++++---- ------+- --+-----| ok p=0.506836
Q7:            |.111-110 01.010.0 0101-110 1101.011| ok p=0.735352
Q8:            |11110110 0101000+ -0101111 0-100111| ok p=0.0507812
Q9:            |...-1... .-.....1 0..1+... .1....^.| ok p=0.522461
Q10:           |...0...0 ^0.....0 1..+0... .0....-.| ok p=0.895508
Q11:           |..0+..^0 -1...^.. ...01... ......1.| ok p=0.822266
Q12:           |.001..-+ 0....-.. .111.... ......1.| ok p=1
Q13:           |.1-1..0- 1....0.. 100....1 ....1...| ok p=1
Q14:           |.-+...10 .....0.. 1-+....1 ....1...| ok p=0.556641
Q15:           |.0+....0 ........ +01....+ ....-...| ok p=0.998047
Q16:           |.^+..... .0...... .^^..... ........| ok p=0.892578
Q17:           |..1..... .1....0. ^......^ ....^...| ok p=1
Q18:           |..0..... .-....1. ........ ........| ok p=0.999023
Q19:           |........ ......-. ........ ........| ok p=0.860352
Q20:           |0....... .^...... ........ ........| ok p=1
Q21:           |0....... ......^. ........ ........| ok p=0.485352
Q22:           |+....... ........ ........ ........| ok p=0.501953
Q23:           |........ ........ ........ ........| ok p=1
Q24:           |^....... ........ ........ ........| ok p=1
----------------------------------------------------------------
Q25-32:        |........ ........ ........ ........| ok p=1
----------------------------------------------------------------
Q33:           |0....... ........ ........ ........| ok p=1
Q34:           |1....... ........ ........ ........| ok p=0.498047
----------------------------------------------------------------
Q35-59:        |±....... ........ ........ ........| ok p=1
----------------------------------------------------------------
Q60:           |+.....0. ........ .....00. ........| ok p=1
Q61:           |+.....1. ........ 11....1. ........| ok p=0.525391
Q62:           |-.....-. ........ 10...-+. ........| ok p=0.493164
Q63:           |+.....-. ........ +-...?-. ........| ok p=0.50293
Q64:           |....-++. ........ +-....-. ..-.+..+|
```

differential path 4

Yet unknown chosen-prefix collision attack

1. Other differential path family
   - Same message differences for all 4 near-collision attacks (up to sign)
     $\delta m_4 = 2^{31},\ \delta m_{11} = \pm 2^{15},\ \delta m_{14} = 2^{31}$ (same as [WY04])
   - No systematic elimination using $\delta m_{11} = \pm 2^b$ as in [SSA+09]

2. Yet unknown birthday search
   - Birthday search is preprocessing phase to find differences that can be cancelled by differential path family
   - Less flexible differential path family ⇒ Higher complexity birthday search
   - Approx. 2^{49} MD5 compression function calls
   - To compare: our attack in total has average complexity of $2^{44.6}$ MD5 calls

Yet unknown chosen-prefix collision attack

3. Yet unknown differential path construction algorithm
   - Differences in *all* bit positions of Q6 in *all* 4 near-collision attacks
   - Not a characteristic of known construction algorithms
   - Their connection search-space strictly contained in our search-space
   - Initial tests show this approach to be significantly slower than our approach
   - Probability successful connection over 4 steps drops approx. by factor 2^{-13}

For a more extensive analysis, see http://eprint.iacr.org/2013/358

Part IV – Improvements

To Conclude…

Improving counter-cryptanalysis

- Reduce chance at false negatives
- Need ‘exhaustive’ list of $(\delta B, \delta WS)$

- Case SHA-1
  - Need ‘exhaustive’ list of feasible disturbance vectors
  - Heuristic searches have found only two interesting classes of D.V.s
  - Make selection based on cost function, e.g. [Ste12]

- Case MD5
  - Finding feasible δB unfinished work
  - Literature focused on finding attacks better than Wang et al.’s
  - W.I.P. heuristic ‘exhaustive’ surveys for δB
  - Using cost function determining lower-bound for attack complexity

Improving counter-cryptanalysis
Early stop

Improving counter-cryptanalysis

- Each $(\delta B, \delta WS)$-guess costs 1 full compression function call
- Speed-up by early stop
  - Use very fast pre-check and only do full work with low probability
  - Without introducing possible false negatives
  - Find unavoidable conditions on message and state bits
    (conditions necessary for all possible feasible attacks based on $(\delta B, \delta WS)$)

- Case SHA-1
  - Determine unavoidable message bitrelations over steps 30-70
  - E.g., using exact joint local-collision analysis [Ste12]
  - Expect at least 4 bitrelations per Disturbance Vector
  - Would result in total cost factor < 2 instead of 16

Improving counter-cryptanalysis

- Case MD5 (per δB, δWS_i)
  - More complex, multiple types of pre-checks
  - Do check whether transitions $(0,0,0,0) \leftrightarrow (2^{31}, 2^{31}, 2^{31}, 2^{31})$ happen correctly
  - Determine lower-bound on attack complexity $< 2^{64}$
  - Determine K = # last steps that may vary s/t complexity $< 2^{64}$
  - Do check whether $(2^{31}, 2^{31}, 2^{31}, 2^{31}) \rightarrow (2^{31}, 2^{31}, 2^{31}, 2^{31})$ happen correctly for first 16-K steps of round 4
  - Find unavoidable bitconditions in round 2 & 3
    - i.e., check whether forcing negated bitcondition results in complexity $\geq 2^{64}$

To conclude…

Conclusions

- Migrate away from MD5 and SHA-1 based signature schemes
- The easy answer of "migrate to more secure signature schemes" is not a practical solution for the near future
- Instead allow old signatures, but protect verifiers against forgeries
- Real-time signature forgery detection possible
  - works for collision attacks on MD5 & SHA-1 [Ste12]
  - recovers full differential paths
- Reference implementation available
  - Feedback requested!
  - Let me know where it is used!

Conclusions

- Flame uses chosen-prefix collision attack ‘in the wild’
  - But an entirely new variant!
  - Different differential path family than [SSA+09]
  - Yet unknown birthday search
  - Yet unknown block-wise elimination procedure
  - Yet unknown differential path construction algorithm
  - New attack has higher complexity than [SSA+09]

Open questions

- Who made Flame?
  - Evidence points to world-class cryptanalysts, not just hackers
  - Adds to predominant speculation of nation-state behind Flame

- Why develop a new variant attack before Feb 2010?
  - But our attack implementation is public since June 2009 (see [Ste12])
  - Requires large effort, done in parallel
  - Nevertheless: exposes their cryptanalytic knowledge

Open questions

- Will the first successful SHA-1 attack be due to scientific efforts or not?
  - Recent years have shown almost no public efforts on SHA-1.
  - Flame's attack on MD5 was developed independently and in parallel to public scientific efforts.
  - Perhaps attacks on SHA-1 are as well…
  - Nation-states have more computing resources than academics…

- Counter-cryptanalysis
  - New paradigm against cryptanalytic attacks
  - Collision detection first practical example
  - Can we construct other (practical) examples against known cryptanalytic attacks?

Thank you for your attention

Questions?

References

[DL05] *"The story of Alice and her Boss"*, M. Daum, S. Lucks, June 2005.

[GIS05] *A note on practical value of single hash collisions for special file formats*, M. Gebhardt, G. Illies, W. Schindler, NIST Hash Workshop, 2005.

[Kam04] *MD5 considered to be harmful someday*, Dan Kaminsky, 2004.

[Kas12] *Flame*, Kaspersky Lab, http://www.kaspersky.com/flame

[LdW05] *On the possibility of constructing meaningful hash collisions for public keys*, A.K. Lenstra, B. de Weger ACISP 2005, LNCS Vol. 3574, pp. 267-279, Springer, 2005.

[Mik04] *Practical attacks on Digital Signatures using MD5 Message Digest*, Ondrej Mikle, 2004.

[R91] *MD5*, Ron Rivest, 1991, RFC 1321.

[SLdW07] *Chosen-prefix collisions and colliding X.509 certificates for different identities*, M. Stevens, A.K. Lenstra, B. de Weger, EUROCRYPT 2007, LNCS Vol. 4515, pp. 1-22, Springer, 2007.

[Sot12] *Analyzing the MD5 collision in Flame*, Alex Sotirov, SummerCon conference, New York, June 2012.

[SSA+09] *Short Chosen-Prefix Collisions for MD5 and the Creation of a Rogue CA Certificate*, M. Stevens, A. Sotirov, J. Appelbaum, A.K. Lenstra, D. Molnar, D.A. Osvik, B. de Weger, CRYPTO 2009, LNCS Vol. 5677, pp. 55-69, Springer, 2009.

[Ste12] *Attacks on Hash Functions and Applications*, Marc Stevens, PhD thesis, Leiden University. (See also the open-source project at: http://code.google.com/p/hashclash/)

[WY04] *How to break MD5 and other hash functions*, X. Wang, H. Yu, EUROCRYPT 2005, LNCS Vol. 3494, pp. 19-35, Springer, 2005.