MD5 Considered Harmful Today

Creating a rogue CA certificate

Alexander Sotirov *New York, USA*
Marc Stevens *CWI, Netherlands*
Jacob Appelbaum *Noisebridge/Tor, SF*
Arjen Lenstra *EPFL, Switzerland*
David Molnar *UC Berkeley, USA*
Dag Arne Osvik *EPFL, Switzerland*
Benne de Weger *TU/e, Netherlands*

Introduction

- International team of researchers
  - working on chosen-prefix collisions for MD5
- MD5 is still used by real CAs to sign SSL certificates today
  - MD5 has been broken since 2004
  - theoretical CA attack published in 2007
- We used a MD5 collision to create a rogue Certification Authority
  - trusted by all major browsers
  - allows man-in-the-middle attacks on SSL

Overview of the talk

- Public Key Infrastructure
- MD5 chosen-prefix collisions
- Generating colliding certificates
    - on a cluster of 200 PlayStation 3's
- Impact
- Countermeasures
- Conclusion

Live demo

1. Set your system date to August 2004
    - intentional crippling of our demo CA
    - not a technical limit of the method itself
2. Connect to our wireless network
    - ESSID “MD5 Collisions Inc”
3. Connect to any secure HTTPS website
    - MITM attack
    - check the SSL certificate!

Part I

Public Key Infrastructure

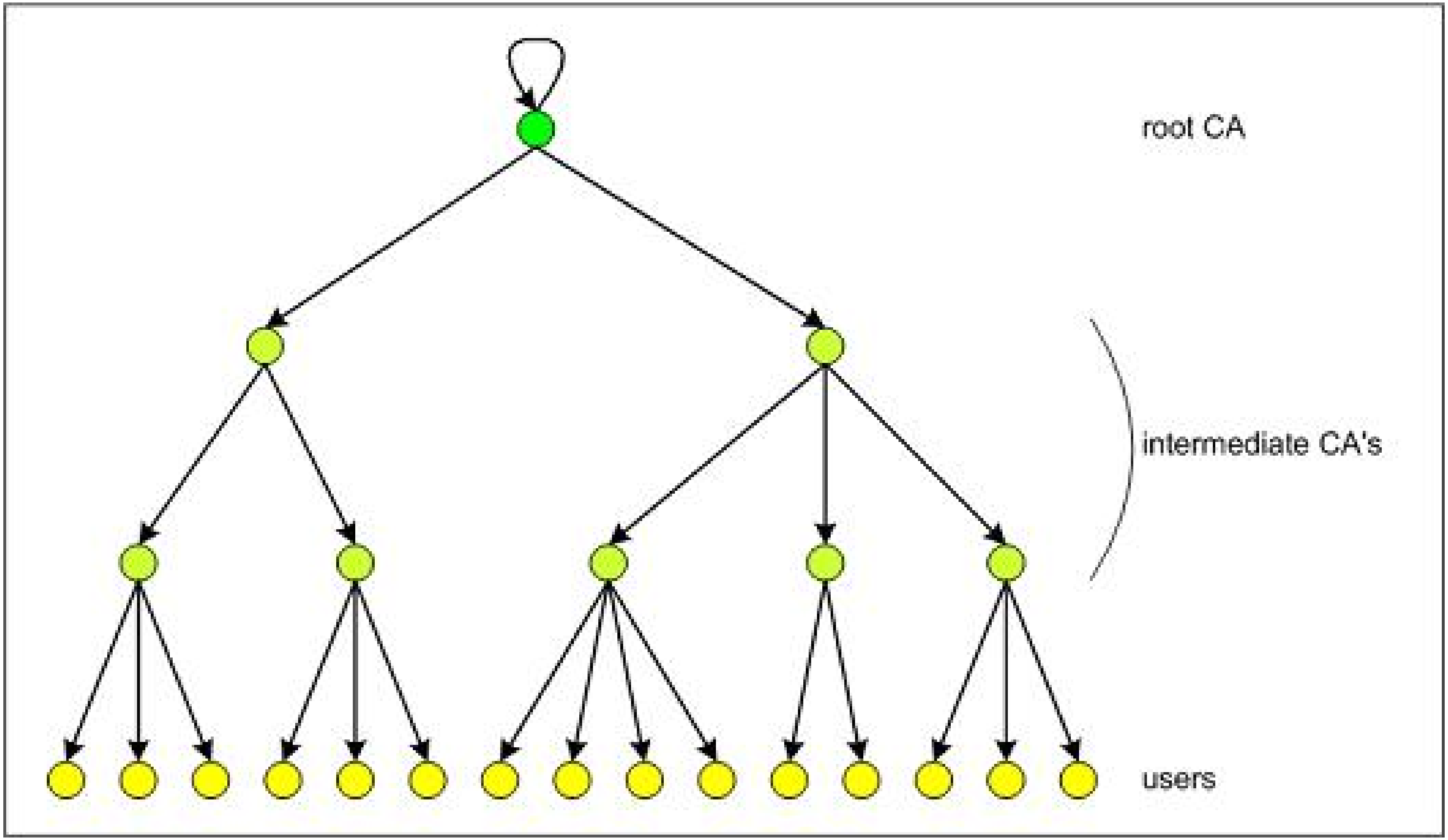
Overview of SSL

- Wide deployment
  - web servers
  - email servers (POP3, IMAP)
  - many other services (IRC, SSL VPN, etc)
- Very good at preventing eavesdropping
  - asymmetric key exchange (RSA)
  - symmetric crypto for data encryption
- Man-in-the-middle attacks
  - prevented by establishing a chain of trust from the website digital certificate to a trusted Certificate Authority

Certification Authorities (CAs)

- Website digital certificates must be signed by a trusted Certificate Authority
- Browsers ship with a list of trusted CAs
  - Firefox 3 includes 135 trusted CA certs
- CAs’ responsibilities:
  - verify the identity of the requestor
  - verify domain ownership for SSL certs
  - revoke bad certificates

Certificate hierarchy



Obtaining certificates

1. User generates private key
2. User creates a Certificate Signing Request (CSR) containing
   - user identity
   - domain name
   - public key
3. CA processes the CSR
   - validates user identity
   - validates domain ownership
   - signs and returns the certificate
4. User installs private key and certificate on a web server

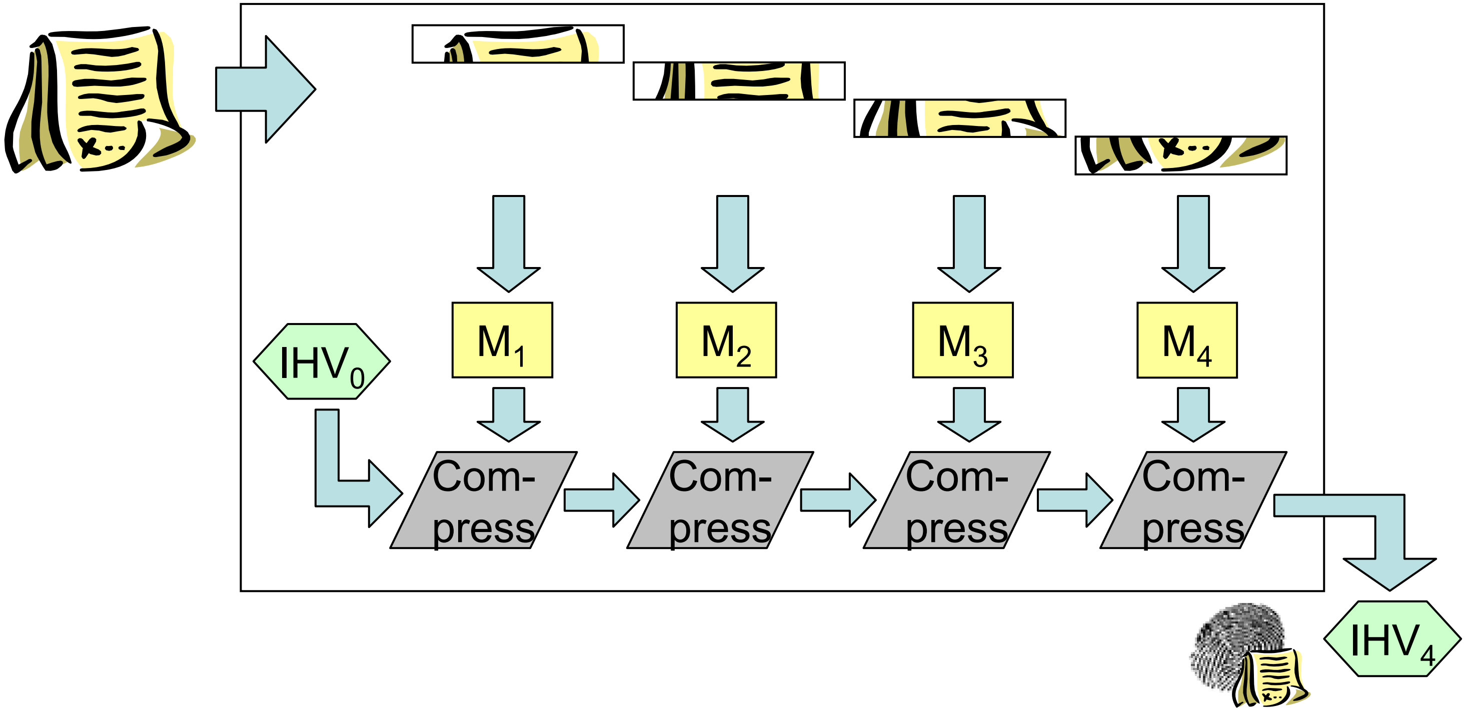
Part II

MD5 Collisions

Overview of MD5

Hash function MD5 designed in 1991:

- Iterative design using compression function



Collisions = different messages, same hash

MD5 Collisions in 2004

2004: First MD5 collision attack

- Only difference between messages in random looking 128 collision bytes
- Currently < 1 second on PC

MD5( ) = MD5( )

MD5 Collisions in 2004

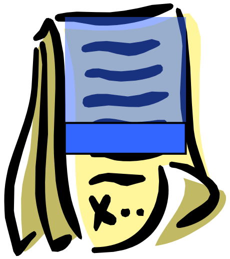
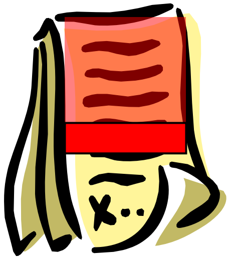
Attack scenarios

- Generate specific collision blocks
- Use document format IF...THEN...ELSE
- Both payloads present in both files
- Colliding PostScript files with different contents
- Similar examples with other formats: DOC, PDF
- Colliding executables with different execution flows

MD5 Collisions in 2007

2007: Stronger collision attack

- *Chosen-Prefix Collisions*
- Messages can differ freely
  up to the random looking 716 collision bytes
- Currently approx. 1 day on PS3+PC

MD5() = MD5()

MD5 Collisions in 2007

Second generation attack scenarios

- Using chosen-prefix collisions
- No IF...THEN...ELSE necessary
  - Each file contains single payload instead of both
  - Collision blocks not actively used in format
- Colliding executables
  - Malicious payload cannot be scanned in harmless executable
- Colliding documents (PDF, DOC, ...)
  - Collision blocks put inside hidden raw image data

Part III

Generating Colliding Certificates

History of colliding certificates

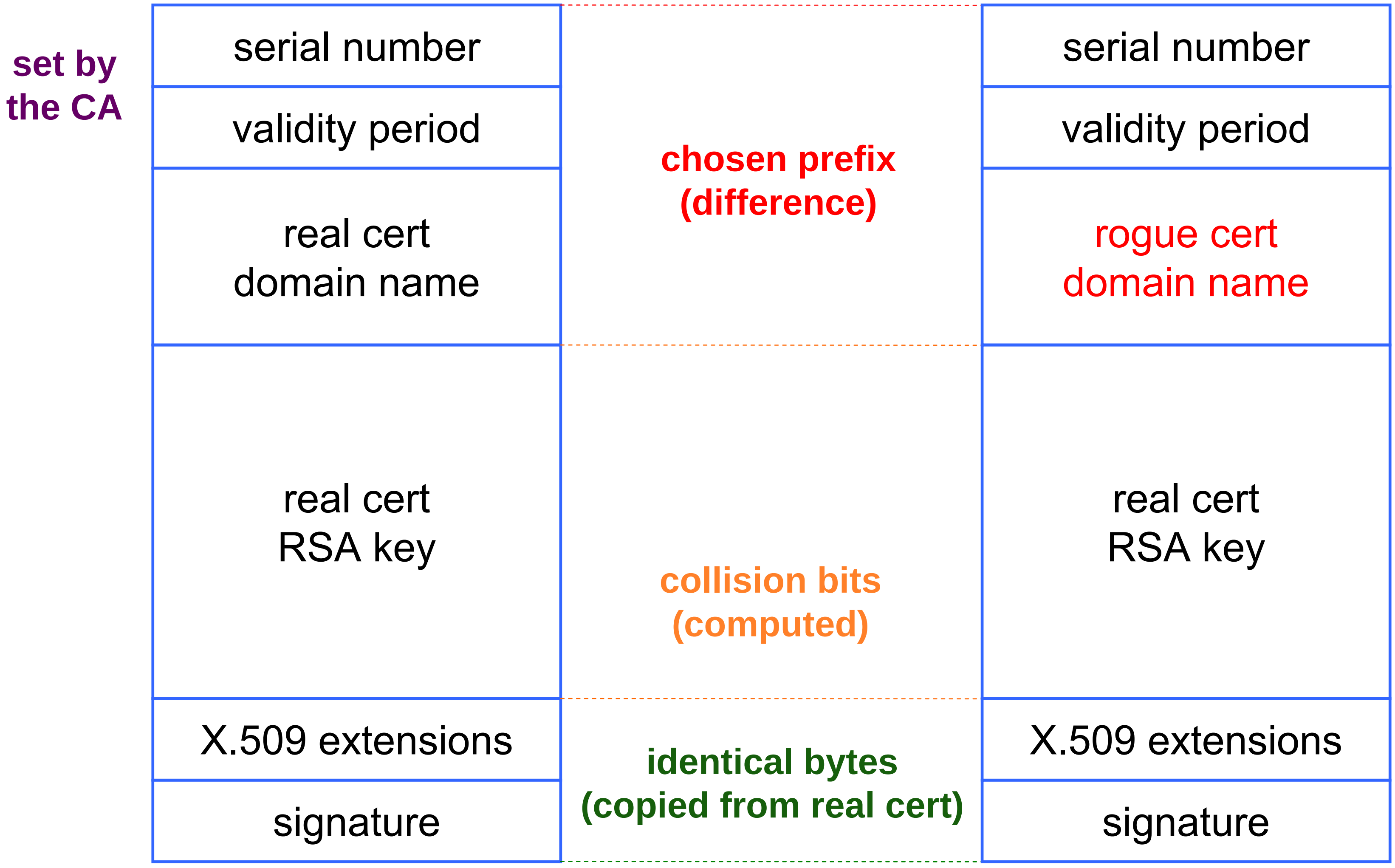
Certificates with colliding to-be-signed parts

- generate a pair of certificates
- sign the legitimate certificate
- copy the signature into the rogue cert

Previous work

- Different RSA public keys in 2005
  - using 2004 collision attack
- Different identities in 2006
  - using chosen-prefix collisions
  - the theory is well known since 2007

Colliding certificates in 2006



set by the CA

serial number

validity period

real cert domain name

real cert RSA key

X.509 extensions

signature

chosen prefix (difference)

collision bits (computed)

identical bytes (copied from real cert)

serial number

validity period

rogue cert domain name

real cert RSA key

X.509 extensions

signature

Vulnerable CAs in 2008

- We collected 30,000 website certificates
  - 9,000 of them were signed with MD5
  - 97% of those were issued by RapidSSL
- CAs still using MD5 in 2008:
  - RapidSSL
  - FreeSSL
  - TrustCenter
  - RSA Data Security
  - Thawte
  - verisign.co.jp

Predicting the validity period

- RapidSSL uses a fully automated system
- The certificate is issued exactly 6 seconds after we click the button and expires in one year.

Predicting the serial number

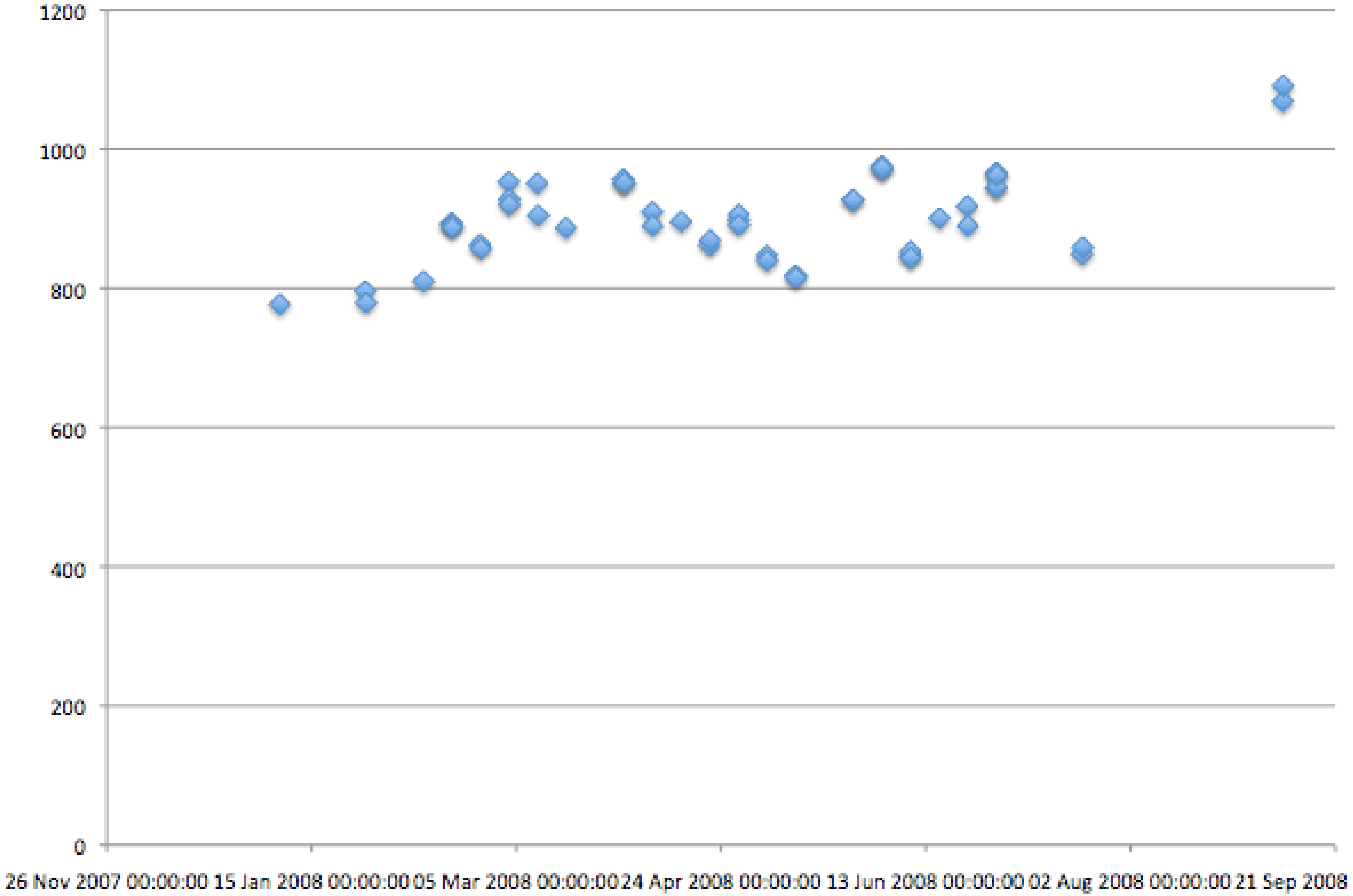
RapidSSL uses sequential serial numbers:

```
Nov  3 07:42:02 2008 GMT   643004
Nov  3 07:43:02 2008 GMT   643005
Nov  3 07:44:08 2008 GMT   643006
Nov  3 07:45:02 2008 GMT   643007
Nov  3 07:46:02 2008 GMT   643008
Nov  3 07:47:03 2008 GMT   643009
Nov  3 07:48:02 2008 GMT   643010
Nov  3 07:49:02 2008 GMT   643011
Nov  3 07:50:02 2008 GMT   643012
Nov  3 07:51:12 2008 GMT   643013
Nov  3 07:51:29 2008 GMT   643014
Nov  3 07:52:02 2008 GMT     ?
```

Predicting the serial number

- Remote counter
  - increases only when people buy certs
  - we can do a query-and-increment operation at a cost of buying one certificate
- Cost
  - $69 for a new certificate
  - renewals are only $45
  - up to 20 free reissues of a certificate
  - $2.25/query-and-increment operation

Certificates issued per weekend



Predicting the serial number

1. Get the serial number S on Friday
2. Predict the value for time T on Sunday to be S+1000
3. Generate the collision bits
4. Shortly before time T buy enough certs to increment the counter to S+999
5. Send colliding request at time T and get serial number S+1000

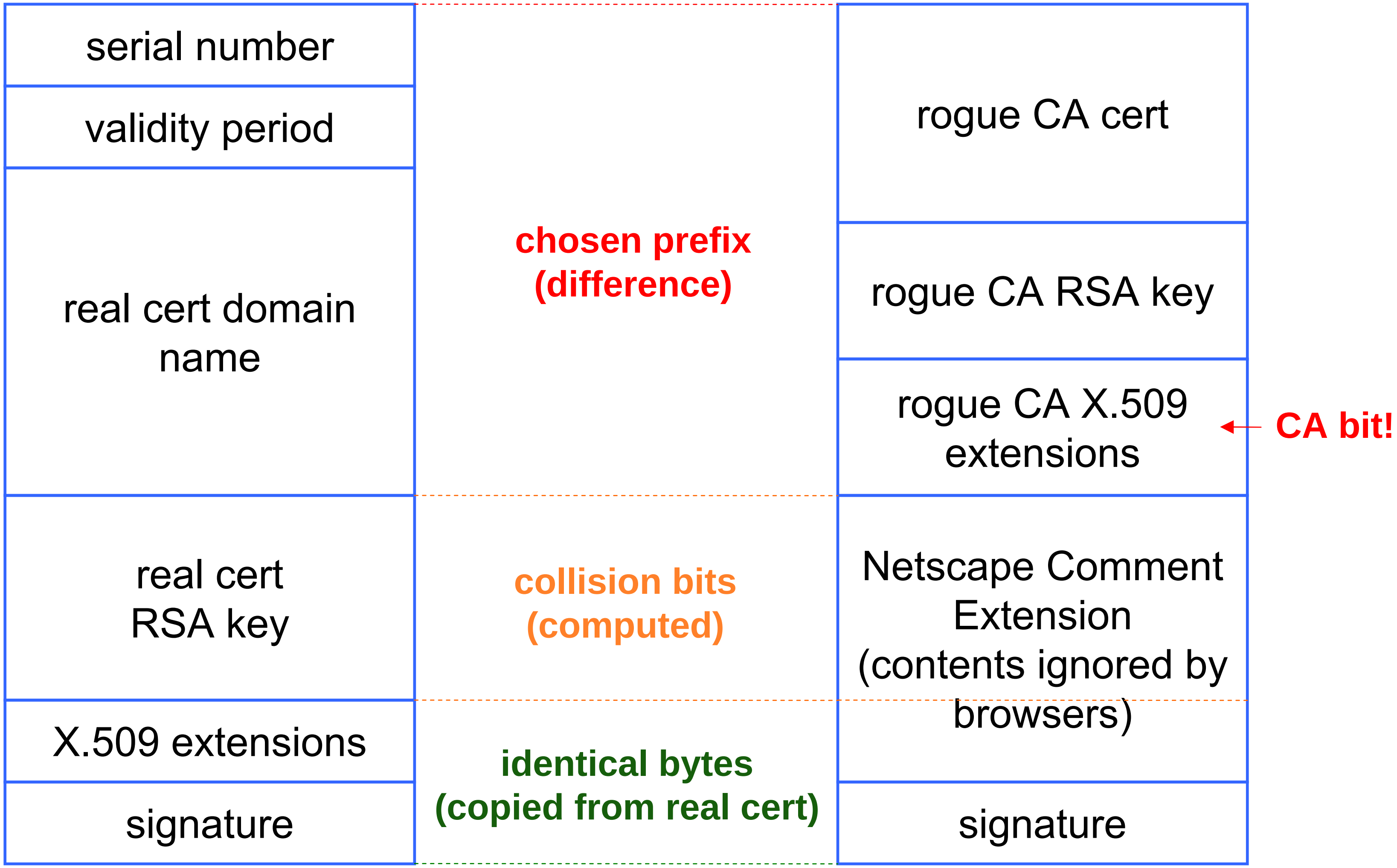
Collision generation

Based on the 2007 chosen-prefix collisions paper with new improvements

1-2 days on a cluster of 200 PlayStation 3’s

Equivalent to 8000 desktop CPU cores or $20,000 on Amazon EC2

Creating an intermediate CA



serial number

validity period

real cert domain name

real cert RSA key

X.509 extensions

signature

chosen prefix (difference)

collision bits (computed)

identical bytes (copied from real cert)

rogue CA cert

rogue CA RSA key

rogue CA X.509 extensions

CA bit!

Netscape Comment Extension (contents ignored by browsers)

signature

Real life execution of the attack

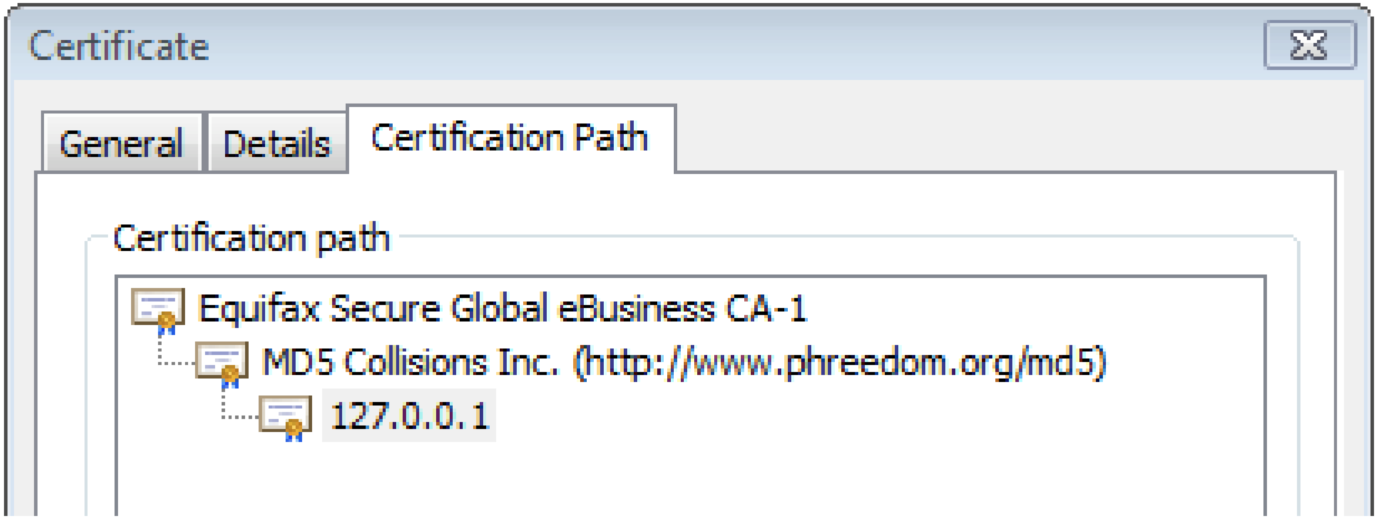
- 3 failed attempts
  - problems with timing
  - other CA requests stealing our serial number
- Finally success on the 4th attempt!
- Total cost of certificates:
  USD $657

Part IV

Impact

Man-In-The-Middle

- We can sign fully trusted certificates
- Perfect man-in-the-middle attacks



Equifax Secure Global eBusiness CA-1
MD5 Collisions Inc. (http://www.phreedom.org/md5)
127.0.0.1

- A malicious attacker can pick a more realistic CA name and fool even experts

Connection hijacking

MITM requires connection hijacking:

- Insecure wireless networks
- ARP spoofing
- Proxy autodiscovery
- DNS spoofing
- Owning routers

Part V

Countermeasures

Preventing harm from our cert

- We’re not releasing the private key
- Our CA cert was backdated to Aug 2004
  - just for demo purposes, a real malicious attacker can get a cert that never expires
- Browser vendors can blacklist our cert
  - we notified them in advance
- Users *might* be able to blacklist our cert

Revocation issues

Our CA cert is not easily revocable!

- CRL and OCSP get the revocation URL from the cert itself
- Our cert contains no such URL
- Revocation checking is disabled in Firefox 2 and IE6 anyways

Possible fixes: Large organizations can set up their own custom OCSP server and force OCSP revocation checking.

EV certs

Extended Validation (EV) certs:

- supported by all major browsers
- EV CAs are not allowed to use MD5
- safe against this attack

Do users really know how to tell the difference between EV and regular certs?

Repeating the attack

With optimizations the attack might be done for $2000 on Amazon EC2 in 1 day

We want to prevent malicious entities from repeating the attack:

- We are not releasing our collision finding implementation or improved methods until we feel it’s safe
- We’ve talked to the affected CAs: they will switch to SHA-1 very, very soon

Has this already been done?

No way to tell.

- The theory has been public since 2007
- Our legitimate certificate is completely innocuous, the collision bits are hidden in the RSA key, but they look random

Can we still trust CA certs that have been used to sign anything with MD5 in the last few years?

Lessons for the future

- We need defense in depth
  - random serial numbers
  - random delay when signing certs
- Future challenges:
  - second preimage against MD5
  - collisions in SHA-1
- Dropping support for a broken crypto primitive is very hard in practice
  - but crypto can be broken overnight
  - what do we do if SHA-1 or RSA falls tomorrow?

Part VI

Conclusion

Conclusion

- No need to panic, the Internet is not *completely* broken
- The affected CAs are switching to SHA-1
- Making the theoretical possible is sometimes the only way you can affect change and secure the Internet

Acknowledgements

The Electronic Frontier Foundation

Jennifer Granick, Joseph Gratz

Our lawyers from CWI, TU/e and EPFL

and all other lawyers we’ve forgotten

Dan Kaminsky for his SSL cert collection

Ralf-Philipp Weinmann for his inspiration

Len Sassaman, Meredith Patterson

Microsoft

Mozilla

SNF, NWO