Investigating PKI: the OpenPGP Web of Trust, With a Side Order of X.509

Ralph Holz, Alexander Ulrich,
Lothar Braun, Nils Kammenhuber

Netzarchitekturen und Netzdienste
Technische Universität München

16 September 2011

OpenPGP and X.509 are PKIs

PKI: Public Key Infrastructure

- In asymmetric crypto, Alice and Bob have a problem
- Key Distribution Problem
- ‘How can I be sure that this is Bob’s key?’

Certification

- Idea: let a Trusted Third Party (TTP) testify
- Testification = Certification = sign(ID, PK)
- Two major standards: OpenPGP and X.509

The Backbone of Security?

OpenPGP

- ‘Everyone can certify everyone else’
- Web of Trust
- Often used for e-mail

X.509 certificates

- Pre-Internet - ITU standard (X.500 series)
- Idea: one global Trusted Third Mega-Party
- Hierarchy, with Certification Authorities at the top
- X.509 certificates for SSL/TLS, S/MIME

We Found This Intriguing

This started as a hobby in around 2008.

- Rumours of serious problems in X.509
- But how is OpenPGP doing?
- Wanted a good analysis of deployments
- For both OpenPGP and X.509

Set up two research projects

- Do graph analysis on OpenPGP Web of Trust
- Use active scans and passive monitoring on X.509
- First time both presented together

Part I

OpenPGP

Based on paper at ESORICS

ESORICS, September 2011:

- A. Ulrich, R. Holz, P. Hauck, G. Carle: *Investigating the OpenPGP Web of Trust*.

Introducing the Web of Trust

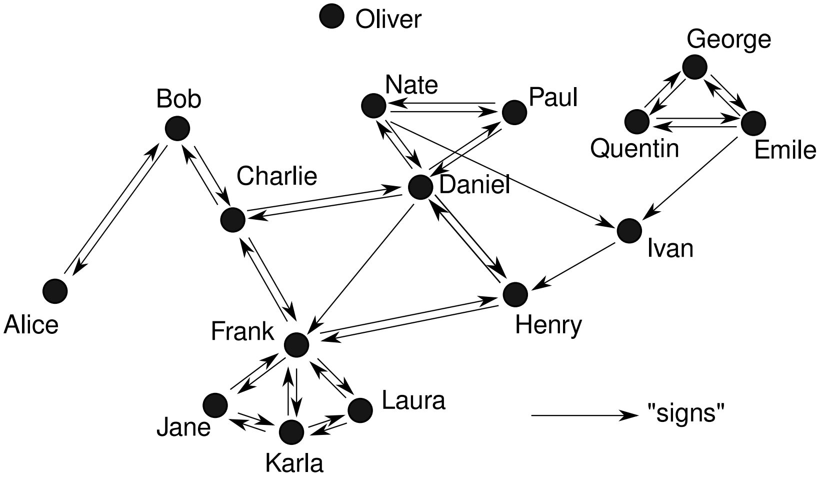
PGP/GnuPG (GPG)

- Widely used implementations of OpenPGP (authentication & encryption)
- Often used for e-mail

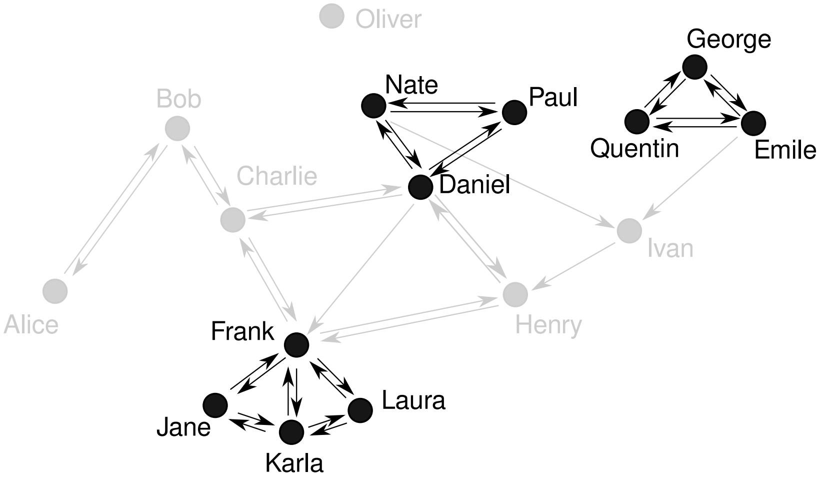
Web of Trust (WoT)

- PKI: everyone can certify anyone else
- Decentralized
- Certification Authorities (CAs) allowed: just very active users

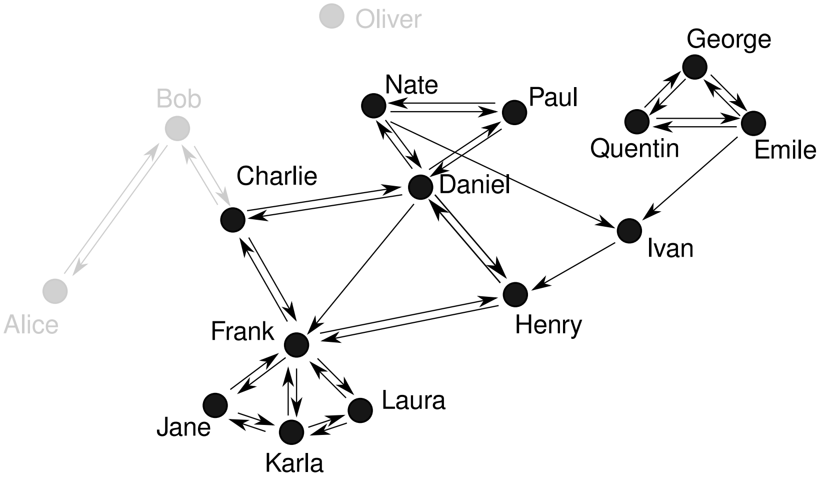
Web of Trust (WoT): Directed Graph



Web of Trust (WoT): Directed Graph



Web of Trust (WoT): Directed Graph



Our Questions (Problem Statement)

Analyze the Web of Trust's graph w.r.t.

- Macro structure
  - How can users profit from the WoT?
- Usefulness to users
  - How effectively can the WoT used?
- Robustness
  - How does the WoT react to changes?
- Further Aspects
  - Social structures? Crypto algorithms?

Our Questions (Problem Statement)

Analyze the Web of Trust's graph w.r.t.

- Macro structure
  - How can users profit from the WoT?
- Usefulness to users
  - How effectively can the WoT used?
- Robustness
  - How does the WoT react to changes?
- Further Aspects
  - Social structures? Crypto algorithms?

Our Questions (Problem Statement)

Analyze the Web of Trust's graph w.r.t.

- Macro structure
  - How can users profit from the WoT?
- Usefulness to users
  - How effectively can the WoT used?
- Robustness
  - How does the WoT react to changes?
- Further Aspects
  - Social structures? Crypto algorithms?

Our Questions (Problem Statement)

Analyze the Web of Trust's graph w.r.t.

- Macro structure
  - How can users profit from the WoT?
- Usefulness to users
  - How effectively can the WoT used?
- Robustness
  - How does the WoT react to changes?
- Further Aspects
  - Social structures? Crypto algorithms?

Our Questions (Problem Statement)

Analyze the Web of Trust's graph w.r.t.

- Macro structure
  - How can users profit from the WoT?
- Usefulness to users
  - How effectively can the WoT used?
- Robustness
  - How does the WoT react to changes?
- Further Aspects
  - Social structures? Crypto algorithms?

Background: OpenPGP

Certification

- Public/private key pair: `pub 2048R/69B003EF`
- User ID: `[Ralph Holz, <holz@net.in.tum.de>]`
- Issue a certificate = `sign(User ID, public key)`

Web of Trust (WoT)

- Network of key servers to upload keys
- Synchronizing Keyservers (SKS) protocol
- Complete history of the network
  (SKS knows no ‘delete’ operation!)

Trust in OpenPGP

Owner Trust

- Alice: “I trust Bob [*very much*/*somewhat*/*not*] to properly identify a person before signing.”
- Private assessment – stored *locally*

Valid keys in GnuPG default settings

- Path length ≤ 5
- Either ‘full’ trust in all owners on path
- Or ≥ 3 distinct paths with ‘marginal’ trust in owners

Deriving Requirements

A good WoT should...

- have certification paths between many (all) keys
  - else it is not useful
- have short certification paths
  - less entities to trust
  - chances of accurately assessing key authenticity
- have redundant paths between keys
  - beneficial for GnuPG trust metric
- be robust
  - removal of a key must have little impact on reachability
- capture social relations between users well
  - trust assessment is easier in communities

Deriving Requirements

A good WoT should...

- have certification paths between many (all) keys
  - else it is not useful
- have short certification paths
  - less entities to trust
  - chances of accurately assessing key authenticity
- have redundant paths between keys
  - beneficial for GnuPG trust metric
- be robust
  - removal of a key must have little impact on reachability
- capture social relations between users well
  - trust assessment is easier in communities

Deriving Requirements

A good WoT should...

- have certification paths between many (all) keys
  - else it is not useful
- have short certification paths
  - less entities to trust
  - chances of accurately assessing key authenticity
- have redundant paths between keys
  - beneficial for GnuPG trust metric
- be robust
  - removal of a key must have little impact on reachability
- capture social relations between users well
  - trust assessment is easier in communities

Deriving Requirements

A good WoT should...

- have certification paths between many (all) keys
  - else it is not useful
- have short certification paths
  - less entities to trust
  - chances of accurately assessing key authenticity
- have redundant paths between keys
  - beneficial for GnuPG trust metric
- be robust
  - removal of a key must have little impact on reachability
- capture social relations between users well
  - trust assessment is easier in communities

Deriving Requirements

A good WoT should...

- have certification paths between many (all) keys
  - else it is not useful
- have short certification paths
  - less entities to trust
  - chances of accurately assessing key authenticity
- have redundant paths between keys
  - beneficial for GnuPG trust metric
- be robust
  - removal of a key must have little impact on reachability
- capture social relations between users well
  - trust assessment is easier in communities

Deriving Requirements

A good WoT should...

- have certification paths between many (all) keys
  - else it is not useful
- have short certification paths
  - less entities to trust
  - chances of accurately assessing key authenticity
- have redundant paths between keys
  - beneficial for GnuPG trust metric
- be robust
  - removal of a key must have little impact on reachability
- capture social relations between users well
  - trust assessment is easier in communities

Let’s Start: Obtaining Our Dataset

Used Dataset

Obtained full snapshot of SKS database

- Stored relevant key properties in SQL DB
- Snapshot contains complete history of network
- Time stamps of key creation, signatures, expiry, revocations, …

Resulting Key Set

Many keys available on the servers

All keys	2.7 millions
Expired, revoked, broken keys	570,000

But not many used for signatures

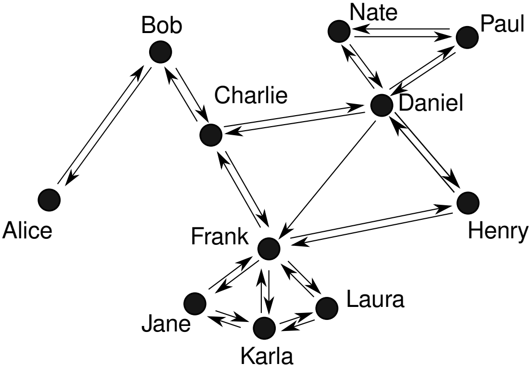
Keys with incoming or outgoing signatures	325,000
Resulting signatures	817,000

Majority of available keys are not verifiable:
no signature chains.

Macro Structure

Macro Structure

Strongly Connected Components (SCCs)



Within an SCC, there is ≥ 1 signature chain between any key pair.

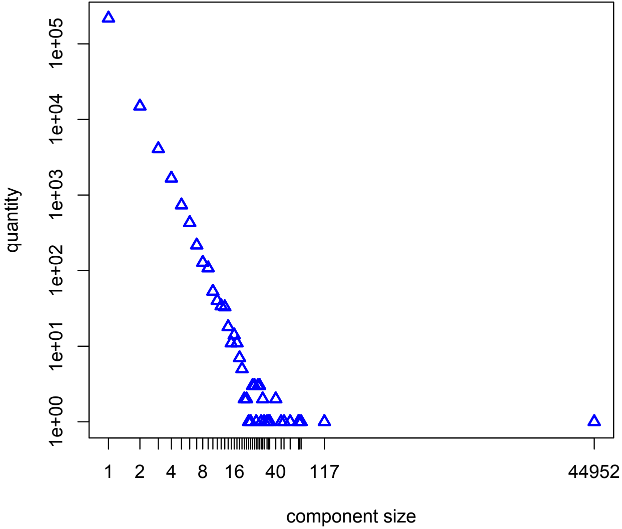
Macro Structure

SCCs are important:
mutual authentication only within the same SCC

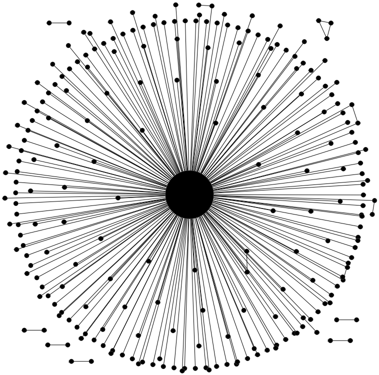
SCCs in the Web of Trust

- Largest SCC (LSCC) of just **45,000** keys (!)
- But there are **240,283** SCCs...
- ... $>$ 100,000 are single nodes (trivial sub-graphs)
- ... $\approx$ 10,000 node pairs

Macro Structure: SCC Sizes



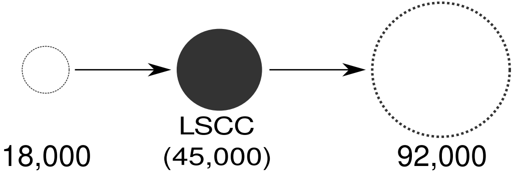
Macro Structure: SCCs and LSCC



SCCs of size > 8 – LSCC in the middle

Macro Structure: Pecularities

Links in/out of LSCC (uni-directional!)



Certification Authorities

- Prominent: Heise, CACert and DFN-Verein (4,200 keys signed in LSCC)
- Heise signed 21,000 keys outside LSCC, too

Impact on Usability

2.7m keys – just 45,000 really profit from the WoT

Significant user activity only in LSCC

- Ratio edges/nodes in LSCC is 9.85, and in whole WoT 2.51
- Recommendation to new users:
  - Get a signature from someone in the LSCC
  - Get a signature from a CA

Focusing on LSCC

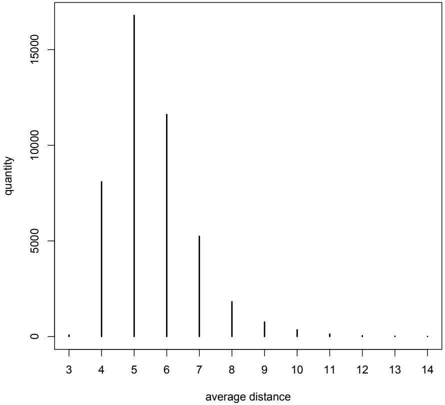
The remainder of this talk will focus on the LSCC

We investigate

- Usefulness (distances, paths, clustering)
- Robustness

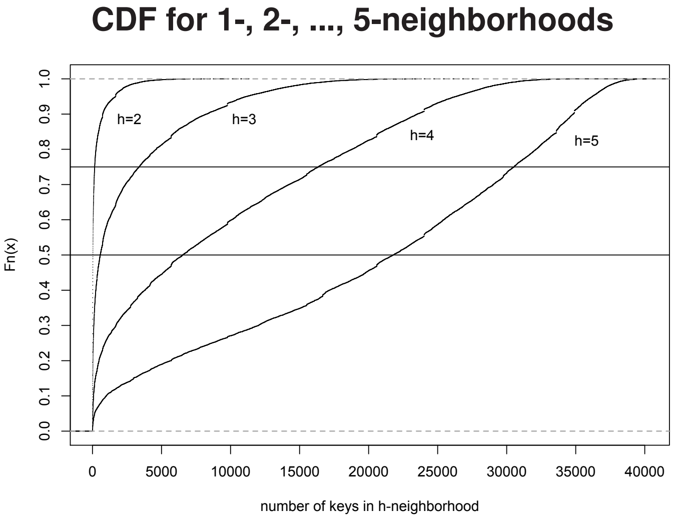
Usefulness:
Distances and Node Degrees

Average Distances in the WoT



- This looked only so-so.
- But it's only the *average* distances.

Nodes reachable via 1,..., 5 hops



Nodes reachable via 1,..., 5 hops

The LSCC is well meshed

- 2-neighborhood (2 hops)
  - Mostly very small neighborhood
  - Very few keys can reach a few hundred keys
- 5-neighborhood (5 hops)
  - 50% chance that a key can reach $\leq$ 22,000 keys
  - Some keys can reach up to almost 38,000 keys

Significance

- Good finding: path lengths not a problem
- But recall: availability of paths is important, too

Node Degrees

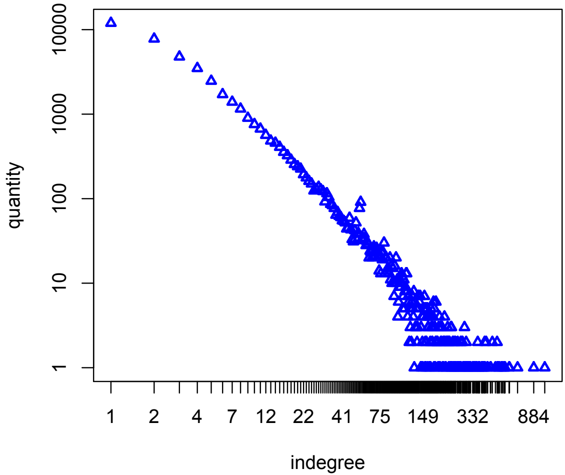
GnuPG views redundant paths as beneficial

- High indegree: key more likely to be verifiable
- High outdegree: higher likeliness of redundant paths

Mutual signatures are also beneficial

- Improves overall verifiability of keys
- Strengthens indegree and outdegree

Node Indegrees



Note: Outdegrees have practically the same distribution

Majority of nodes: low in/outdegree

This is a *bad* finding

- Almost half of keys have indegree 1 or 2
- About $1/3$ of nodes have outdegree 1 or 2
- Mutual signatures: only in 50% of cases...

This means: redundant paths are too rare

- Verify another key: needs direct signatures
- Be verifiable: only via very few other keys

Robustness: Resilience Against Change

Robustness

What happens when keys expire, are revoked, ...

- Paths over these keys become invalid
- Simulated this by randomly removing nodes

Targeted attacks...

- Difficult: either compromise the key...
- ... or delete it on all SKS servers
- Simulated this: remove nodes with high degree first

Is the LSCC a Scale-free Graph?

Scale-free graphs...

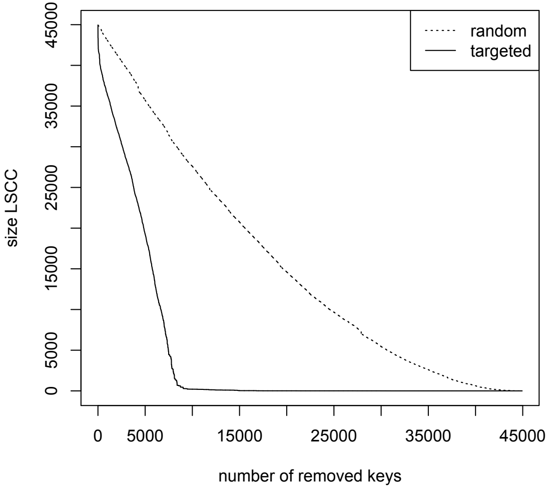
- ... strong hub structure, node degrees follow Power Law
- ... robust against random removal, sensitive to targeted removal of nodes

The LSCC is *not* scale-free

- (Clauset, 2009) recommend Maximum-Likelihood + Kolmogorov-Smirnov test
- The values we obtained rule out Power Law

But similar: many inter-connected hubs

Remove keys, recompute LSCC size



Removing keys

Random removal (expiry, revocation, ...)

- Very robust
- Need to remove $1/3$ of keys to cut LSCC by half

Targeted removal (attack)

- Quite robust – decay not too bad
- Remove all nodes of degree:
  - $>$ 160 (≈ 0.5% of nodes) → LSCC shrinks to 88%
  - $>$ 18 (≈ 11% of nodes) → LSCC shrinks to 50%

Removing keys

Assume CA keys are compromised/revoked

- The LSCC does not care: new size at 94.4%
- Average distances stay the same
- Many paths around the CAs:
  they are not critical components

Key removal is not an efficient attack

- There are *many* hubs, and they are inter-connected
- Not a typical scale-free network

A very good finding for a WoT

Further Aspects

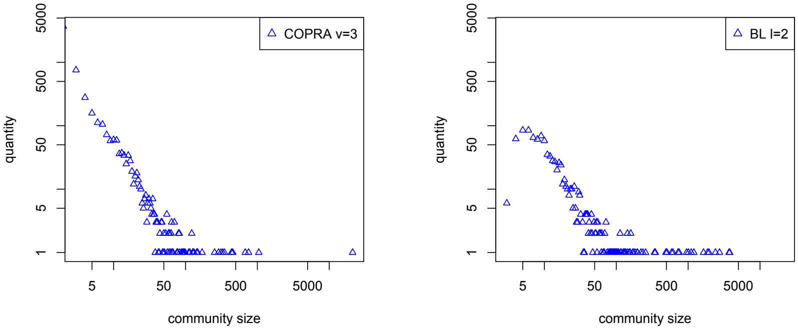
Communities

Analysis of community structure

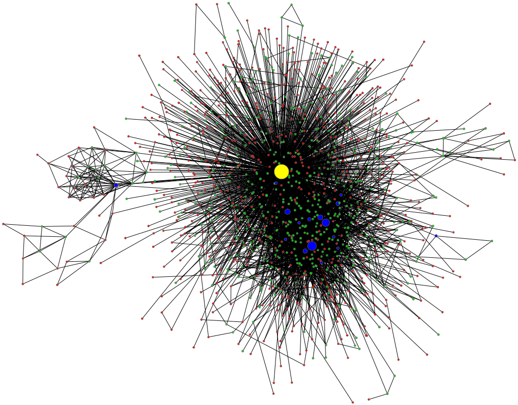
- The LSCC shows a clear Small World Effect
- Used two algorithms for community detection
- Findings:
  - Very strong community structure
  - Communities often dominated by a top-level domain
  - Second-level domains less clearly identifiable
- Details in paper

Distribution of Communities by Size

We tried two methods: COPRA and Blondel et al. (BL)



Communities of size > 5 (COPRA)



Mapping to DNS

Little information in User IDs

- Question: how often are 80% of User IDs in a community in the same TLD?
- Very often: 47%-58%, depending on detection algorithm
- Picture changes entirely for SLDs: only 13%
- A good fraction

Picture changes entirely for SLDs:

- E.g., COPRA: 13%
- Resolution problem?

Tentative Conclusions w.r.t. Communities

Difficult to reach compelling conclusions

- Algorithms agree that pronounced community structure exists
- Mapping to TLDs works OK, but not for SLDs

Consider the huge number of TLDs and SLDs

- Signing process is supported by social links (that’s good)
- Current algorithms too imprecise for better analysis
- Might be worthwhile to follow up on this

Crypto strength

Algorithms in LSCC

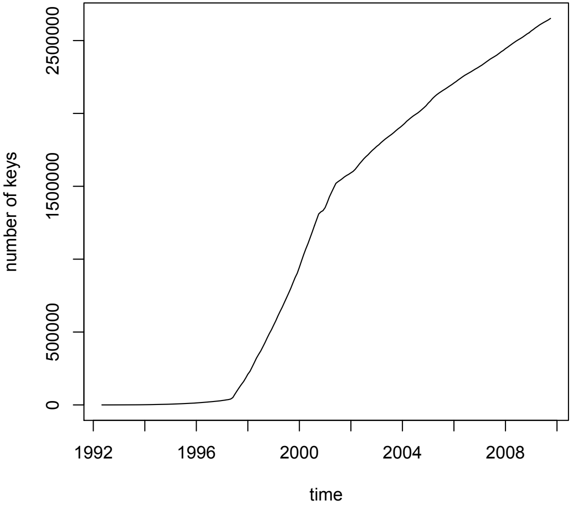
Hash Algorithm	Occ.
SHA1	89.36%
MD5	9.34%
SHA256	1.12%

Key Algorithm	Occ.
DSA-1024	81.32%
RSA-1024	8.68%
RSA-2048	5.36%

Not too much to criticize here

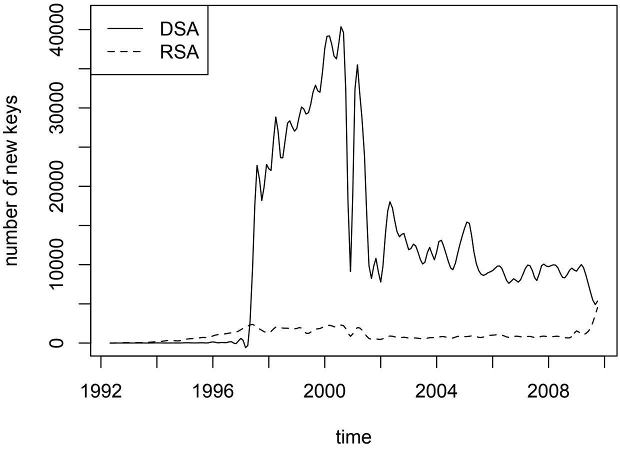
- Some RSA keys of $\leq$ 1,024 bit are well-connected
- Length of < 768 bit occurs ≈ 500 times (problematic)
- 1,024 bit not a problem today, but maybe tomorrow
- Thankfully, few MD5-based signatures

Network History



Network History

RSA and DSA keys



Conclusions

Conclusions

We have found light and dark

- Macro structure
  - ☹ Only users in LSCC really profit from WoT
  - ☺ CAs are useful, but not critical
- Usefulness
  - ☺ Good reachability via $\leq$ 5 hops
  - ☹ Redundant paths too rare!
- Robustness
  - ☺ Very robust against expiration, revocation, ...
  - ☺ Key removal is not an efficient attack

WoT works well in ‘close neighborhoods’ of active nodes – but not otherwise.

Conclusions

We have found light and dark

- Macro structure
  - ☹ Only users in LSCC really profit from WoT
  - ☺ CAs are useful, but not critical
- Usefulness
  - ☺ Good reachability via $\leq$ 5 hops
  - ☹ Redundant paths too rare!
- Robustness
  - ☺ Very robust against expiration, revocation, ...
  - ☺ Key removal is not an efficient attack

WoT works well in 'close neighborhoods' of active nodes – but not otherwise.

Conclusions

We have found light and dark

- Macro structure
  - ☹ Only users in LSCC really profit from WoT
  - ☺ CAs are useful, but not critical
- Usefulness
  - ☺ Good reachability via $\leq$ 5 hops
  - ☹ Redundant paths too rare!
- Robustness
  - ☺ Very robust against expiration, revocation, ...
  - ☺ Key removal is not an efficient attack

WoT works well in ‘close neighborhoods’ of active nodes – but not otherwise.

Conclusions

We have found light and dark

- Macro structure
  - ☹ Only users in LSCC really profit from WoT
  - ☺ CAs are useful, but not critical
- Usefulness
  - ☺ Good reachability via $\leq$ 5 hops
  - ☹ Redundant paths too rare!
- Robustness
  - ☺ Very robust against expiration, revocation, ...
  - ☺ Key removal is not an efficient attack

WoT works well in ‘close neighborhoods’ of active nodes – but not otherwise.

Conclusions

We have found light and dark

- Macro structure
  - ☹ Only users in LSCC really profit from WoT
  - ☺ CAs are useful, but not critical
- Usefulness
  - ☺ Good reachability via $\leq$ 5 hops
  - ☹ Redundant paths too rare!
- Robustness
  - ☺ Very robust against expiration, revocation, ...
  - ☺ Key removal is not an efficient attack

WoT works well in 'close neighborhoods' of active nodes – but not otherwise.

Related Work

Capkun et al., 2001

- LSCC at 12,000 keys only
- Claims Small-World Effect and Power Law distribution

Arenas et al., 2004

- Investigated network as undirected graph
- Degree and community distribution: Power Laws

wotsap

- Continous snapshots and some statistics of LSCC
- Less in-depth; wotsap extraction algorithm is faulty

Part II

X.509 for SSL/TLS

Based on Paper at IMC

Internet Measurement Conference, Berlin 2011:

- R. Holz, L. Braun, N. Kammenhuber, G. Carle: *The SSL landscape – a thorough investigation of the X.509 PKI using active and passive measurements*.

Everyone has heard about DigiNotar.

Right?

That was in 2011.

Our story starts in 2008.

Everyone has heard about DigiNotar.

Right?

That was in 2011.

Our story starts in 2008.

Everyone has heard about DigiNotar.

Right?

That was in 2011.

Our story starts in 2008.

Everyone has heard about DigiNotar.

Right?

That was in 2011.

Our story starts in 2008.

How This Got Our Interest

Early December 2008

- StartSSL.com reports serious flaw in certification process of Comodo CA
- A sub-contractor issued certificates *without identity verification*
- They just debited the credit card
  – and if that worked, it was fine

Q: How seriously do CAs verify identities?

How This Got Our Interest

Christmas 2008

- StartSSL.com Web site becomes victim of hacker
- Hacker obtains certificate for mozilla.com
- StartSSL.com noticed this within an hour and responded with revocation
- But only because they manually double-check requests for high-profile domains...
- The attack was ‘described’ in a report
  – it used an HTTP proxy

Q: How seriously do CAs protect their front/backends?

How This Got Our Interest

February 2009

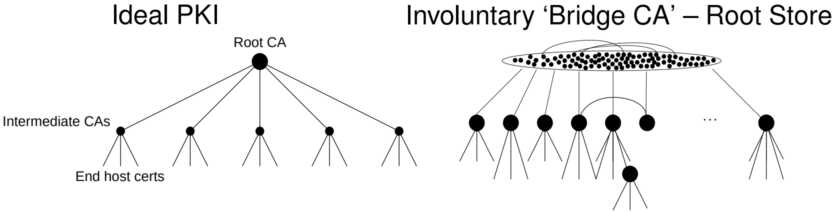
- Paper about ‘easy’ MD5 hash collisions published
- J. Nightingale publishes simple crawling script
- Question: how many MD5-signed certificates are there in the wild?
- Script uses the Alexa Top 1 Million host list for HTTPs crawl

Q: What is the quality of X.509 certificates for SSL?

How This Got Our Interest

State of Mozilla Root Store

- Mozilla in 2009:
  'Does anyone know who owns this root cert?'
- It turned out there were root certs that no-one could remember
- No-one remembered when they got accepted, or why



Root Stores

Remember: your browser chooses the ‘trusted CAs’. Not you.

Mozilla: how to add a root

- File a bug, enter a queue (currently 60 roots waiting)
- Discussion period (public, 1 week)

Followed the mailing list for 2.5 years.

- Never more than 5 people participated actively
- Not one root was rejected in that time

‘Trust’worthy Roots?

CCNIC

- Chinese CA – legal status as independent operator doubtful
- Went through discussion period without delay
- But CCNIC is a known malware distributor
- Caused an outcry by the Chinese Firefox community
- The root was kept. It is also in IE.

Nota bene: Any CA may issue a certificate for any domain. They are all equal.

Removing CCNIC Won’t Help

The EFF has found the following subordinate CAs:

- Department of Homeland Security
- Etisalat
- Booz Allen Hamilton
- Companies: Dell, Ford, Google, Marks and Spencer, Vodaphone

Nota bene: Any CA may issue a certificate for any domain. They are all equal.

The DigiNotar Debacle

Earlier this year: someone hacked Comodo CA

- Issued themselves a few certs
- Browser reaction: blacklist certs, let Comodo live
- Too big too fail?

Two weeks ago, the same person hacked DigiNotar

- Issued themselved 531 certs
- Google, Facebook, Mozilla, CIA, Mossad, Skype
- Attack points to MitM in Iran (ouch)
- For the first time, a Root CA got removed
- Holland says good-bye to their OverheidPKI

PKIScan and PKIMonitor

Since 2009: PKI*Scan*

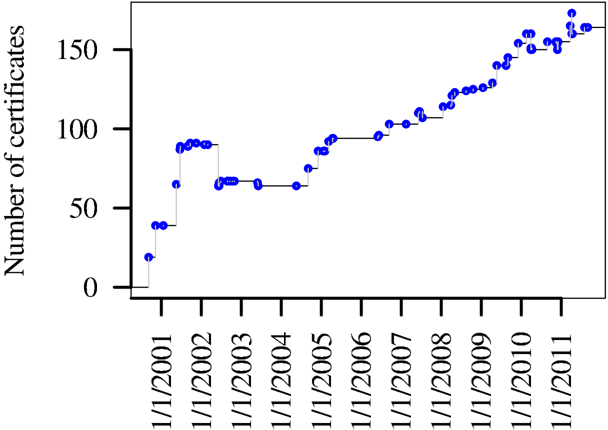
- We scan the Alexa Top 1M list of hosts on port 443
- Store certificates and dump them into a DB
- 8 scans since then (7 used in the paper)

Since 2010: PKI*Monitor*

- We use the 10Gbit monitor at MWN
- Extract certificates right from the session
- 2x 2-week runs since then

We are going to present the results of a *very* thorough analysis at IMC 2011.

The Mozilla Root Store



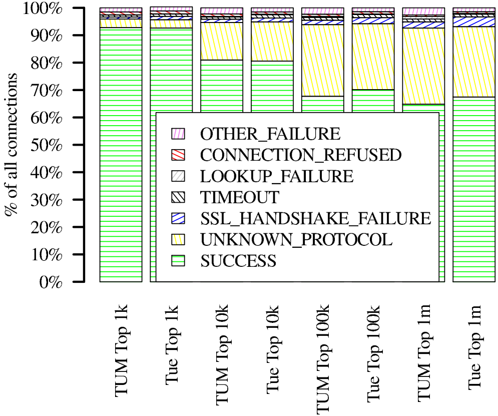
Data Sets

Active scans, monitoring, and EFF

Location	*Time (run)*	*Type*	*Certificates (distinct)*
Tübingen, DE	November 2009	Active scan	833,661 (206,588)
Tübingen, DE	December 2009	Active scan	819,488 (205,700)
Tübingen, DE	January 2010	Active scan	816,517 (204,216)
Tübingen, DE	April 2010	Active scan	816,605 (208,490)
Munich, DE	September 2010	Active scan	829,232 (210,697)
Munich, DE	November 2010	Active scan	827,366 (212,569)
Munich, DE	April 2011	Active scan	829,707 (213,795)
Munich, DE	April 2011	Active scan with SNI	826,098 (212,229)
Shanghai, CN	April 2011	Active scan	798,976 (211,135)
Beijing, CN	April 2011	Active scan	797,046 (211,007)
Melbourne, AU	April 2011	Active scan	833,571 (212,680)
İzmir, TR	April 2011	Active scan	825,555 (211,617)
São Paulo, BR	April 2011	Active scan	833,246 (212,698)
Moscow, RU	April 2011	Active scan	830,765 (213,079)
Santa Barbara, USA	April 2011	Active scan	834,173 (212,749)
Boston, USA	April 2011	Active scan	834,054 (212,805)
Munich, DE	September 2010	Passive monitoring	183,208 (163,072)
Munich, DE	April 2011	Passive monitoring	989,040 (102,329)
EFF servers	March–June 2010	Active IPv4 scan	11,349,678 (5,529,056)

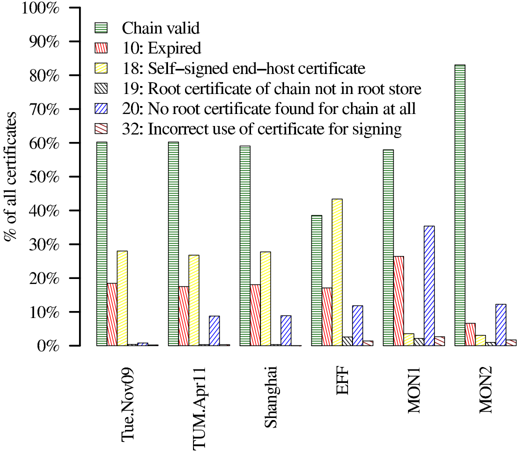
Connection Errors

Scan of all hosts in Nov 2009 and Apr 2011



Connection Errors

Valid chains (no host name check!)



Checking Host Names

Host names in certificate must indicate correct host

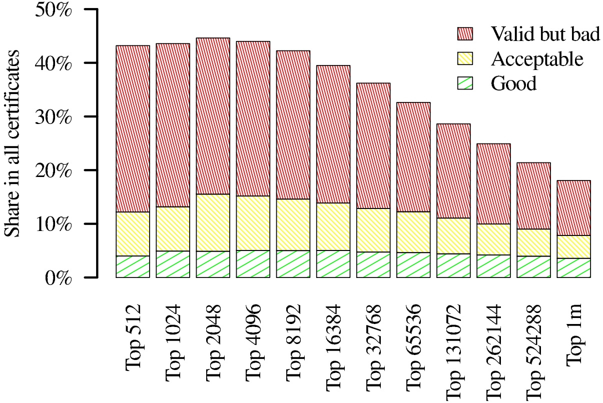
- 18% of certificates have the correct host name *and* a good chain
- Getting slightly better: in Nov 2009, it was 15%
- For 80% of hosts on Alexa list, you get a browser warning
- Server Name Indication (SNI) does not change anything

Unusual host names

- Plesk: 60,000 cases
- localhost: 40,000 cases

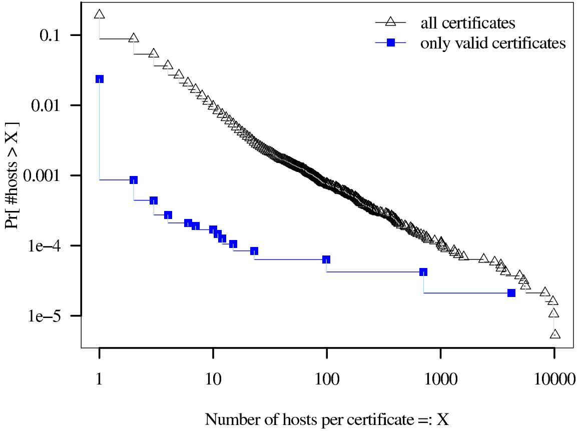
Quality of Certificates

We devised three categories



Reuse of Certificates

Certificates on multiple hosts

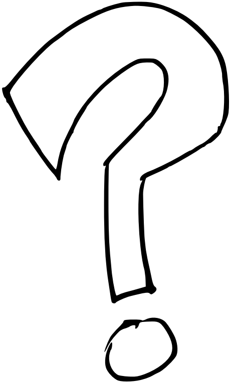


This Was Our Teaser

There's more in the paper

- Crypto: keys and signature algorithms
- Debian weak keys
- Validity periods
- Chain lengths and occurrences
- Self-signed certs
- Issuers
- Differences between locations

Thank You!



Download dataset from `pki.net.in.tum.de`