breaking down the web of trust

seth hardy

22c3: private investigations

29 december 05

before we begin, a question:

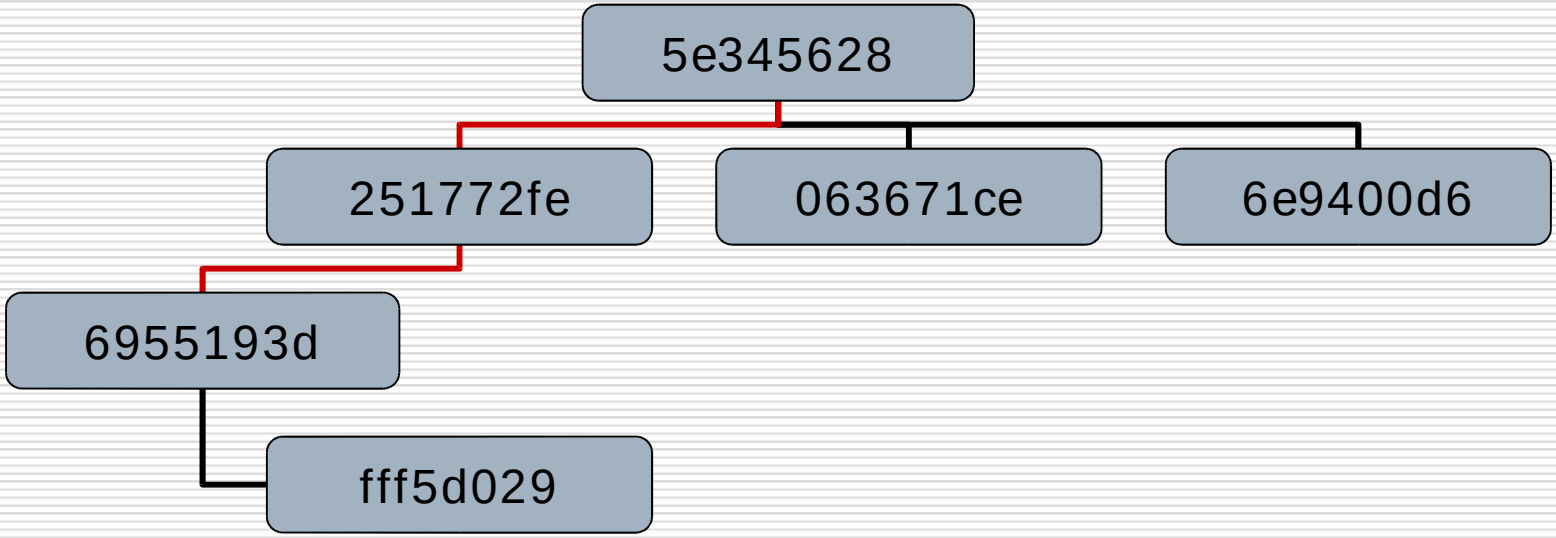
would you sign this key?

```
pub   1024D/1B629B3D 2005-12-27
      Key fingerprint = 965E F829 EA6C 9174 4B46  43E1 4513 9A86 1B62 9B3D
uid                  u1tr4 l4s3r <seekrit@hax0r.com>
sub   2048g/1F8E2EEA 2005-12-27
```

what would you need to know before you did?

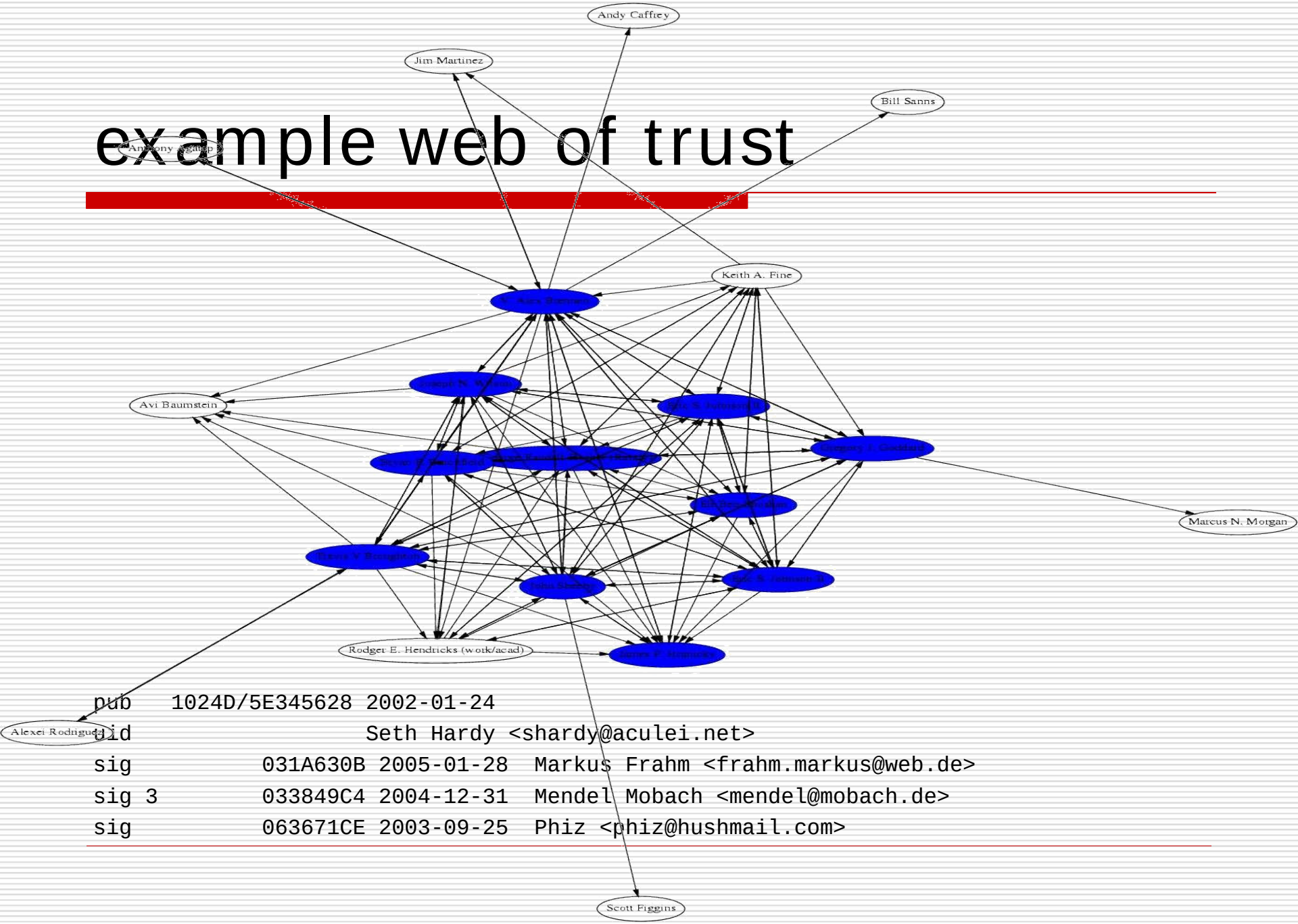
i. the web of trust

why a web of trust?



trust the validity of keys you've never seen before!

example web of trust




```
pub   1024D/5E345628 2002-01-24
uid                  Seth Hardy <shardy@aculei.net>
sig          031A630B 2005-01-28  Markus Frahm <frahm.markus@web.de>
sig 3        033849C4 2004-12-31  Mendel Mobach <mendel@mobach.de>
sig          063671CE 2003-09-25  Phiz <phiz@hushmail.com>
```

web of validation

- signing a key validates the key
  - personal assertion of trustworthiness
- setting trust level is for introductions
  - assigned trust vs. calculated trust
- signed keys are validated, unsigned keys are trusted

building a web of trust

```
$ gpg --update-trustdb
gpg: public key 7FADFC67 is 10809 seconds newer than the signature
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0  valid:   2  signed: 124  trust: 0-, 0q, 0n, 0m, 0f, 2u
No trust value assigned to:
2048R/7FADFC67 2002-05-19
"mike davis (this is a secondary email address since i nolonger control the primary)
     <phar@stonedcoder.org>"
aka "mike davis <phar@thetransmission.net>"
Primary key fingerprint: E2 45 53 28 AF 7E 7D 6F  43 77 E1 F3 92 AD 53 8E

Please decide how far you trust this user to correctly verify other users' keys
(by looking at passports, checking fingerprints from different sources, etc.)

1 = I don't know or won't say
2 = I do NOT trust
3 = I trust marginally
4 = I trust fully
s = skip this key
q = quit
```

Your decision?

validity vs. trust

```
$gpg --edit-key setient
gpg (GnuPG) 1.4.1; Copyright (C) 2005 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.


pub  1024D/251772FE  created: 2004-07-11  expires: never       usage: CS
                     trust: never         validity: full
sub  2048g/C53F06A3  created: 2004-07-11  expires: never       usage: E
[  full  ] (1). Ronald Cotoni (Setient) <miata@sixgirls.org>
```

ii. trust

goals

- verify that a key is accurate
  - check the fingerprint
- verify that key ownership is accurate
  - check the name against photo id
  - send key to email address
- verify the key/identity binding
  - *remember that uids are for human convenience*

key/identity binding?

```
pub   1024D/5E345628 2002-01-24
uid                  Seth Hardy <shardy@aculei.net>
sig          031A630B 2005-01-28  Markus Frahm <frahm.markus@web.de>
sig 3        033849C4 2004-12-31  Mendel Mobach <mendel@mobach.de>
sig          063671CE 2003-09-25  Phiz <phiz@hushmail.com>
```

- signatures are on user ids
- the fingerprint must be checked before *any* user id should be signed
- each user id should be signed separately

i never sign a key that doesn't have a real name on it. there's no way to verify a handle.

verifying a handle is impossible.

who is this person?

verifying a handle is impossible.

how about this guy?

verifying a handle is impossible.

which one is the real thing?

a person only has one unique identity.

going by a pseudonym?

hey baby,
want to sign my key?

```
pub   1024D/1B629B3D 2005-12-27
  Key fingerprint = 965E F829 EA6C 9174 4B46 43E1 4513 9A86 1B62 9B3D
uid   acid burn <acidburn@hackers.com>
sub   2048g/1F8E2EEA 2005-12-27
```

what would you rather it be?

hey baby,
want to sign my key?
i'm not an actress, i promise.

```
pub   1024D/1B629B3D 2005-12-27
  Key fingerprint = 965E F829 EA6C 9174 4B46 43E1 4513 9A86 1B62 9B3D
uid   angelina jolie <ajolie@hackers.com>
sub   2048g/1F8E2EEA 2005-12-27
```

would this be any better?

hey baby,
want to sign my key?
i'm an actress, i promise.
no really, i am, i swear

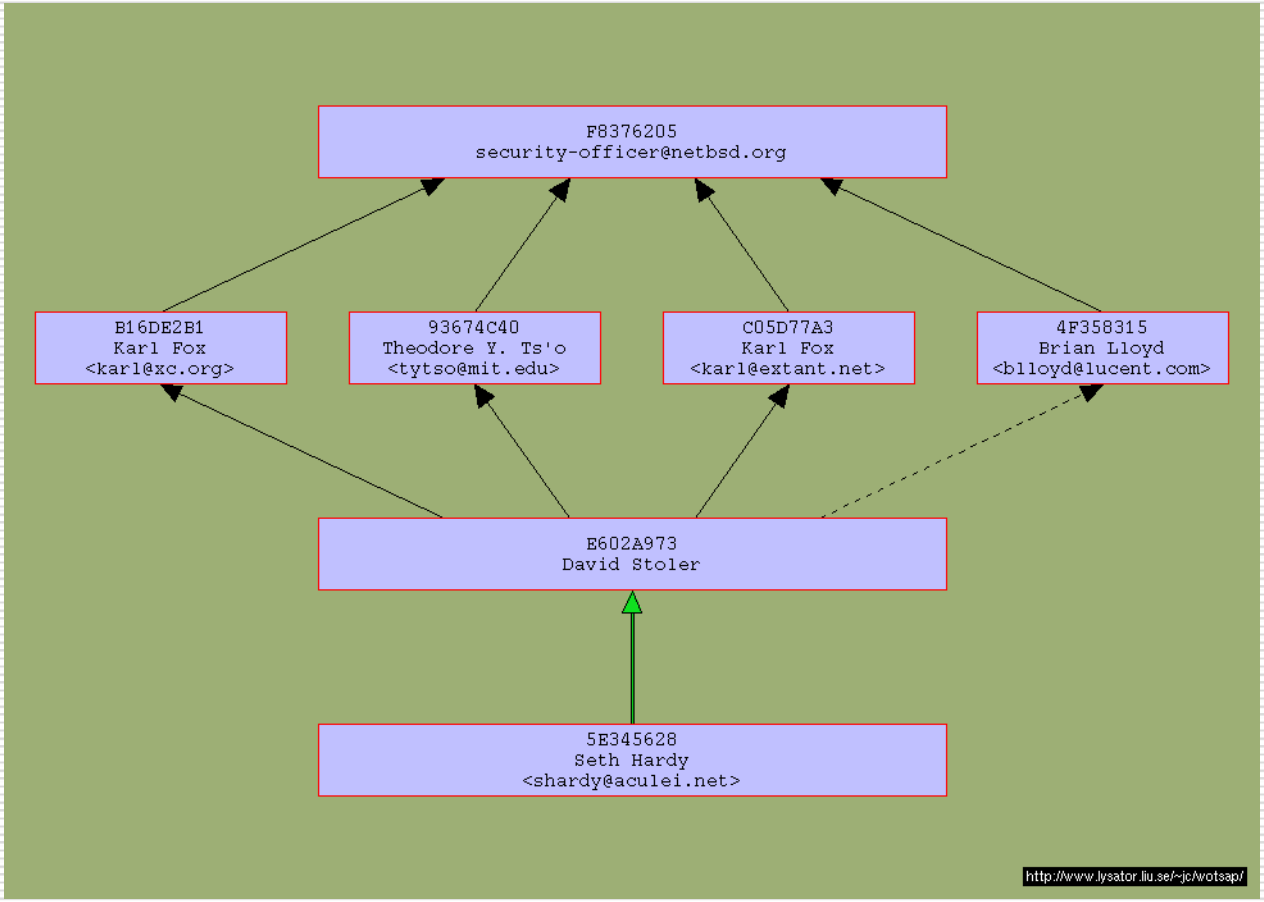
```
pub   1024D/1B629B3D 2005-12-27
  Key fingerprint = 965E F829 EA6C 9174 4B46 43E1 4513 9A86 1B62 9B3D
uid   angelina jolie <ajolie@hackers.com>
sub   2048g/1F8E2EEA 2005-12-27
```

a serious example

- who is security- officer@netbsd.org?
  - they have 24 signatures
  - they have signed 3 other keys
  - msd of 4.6305
  - msd ranking 2750
  - only three hops from my key

```
$ gpg --fingerprint f8376205
pub   1024R/F8376205 1997-07-01
      Key fingerprint = 19 57 B6 26 AB F1 81 A4  A4 F9 4E CE F5 27 4C F5
uid                  security-officer@netbsd.org
```

four paths, three hops



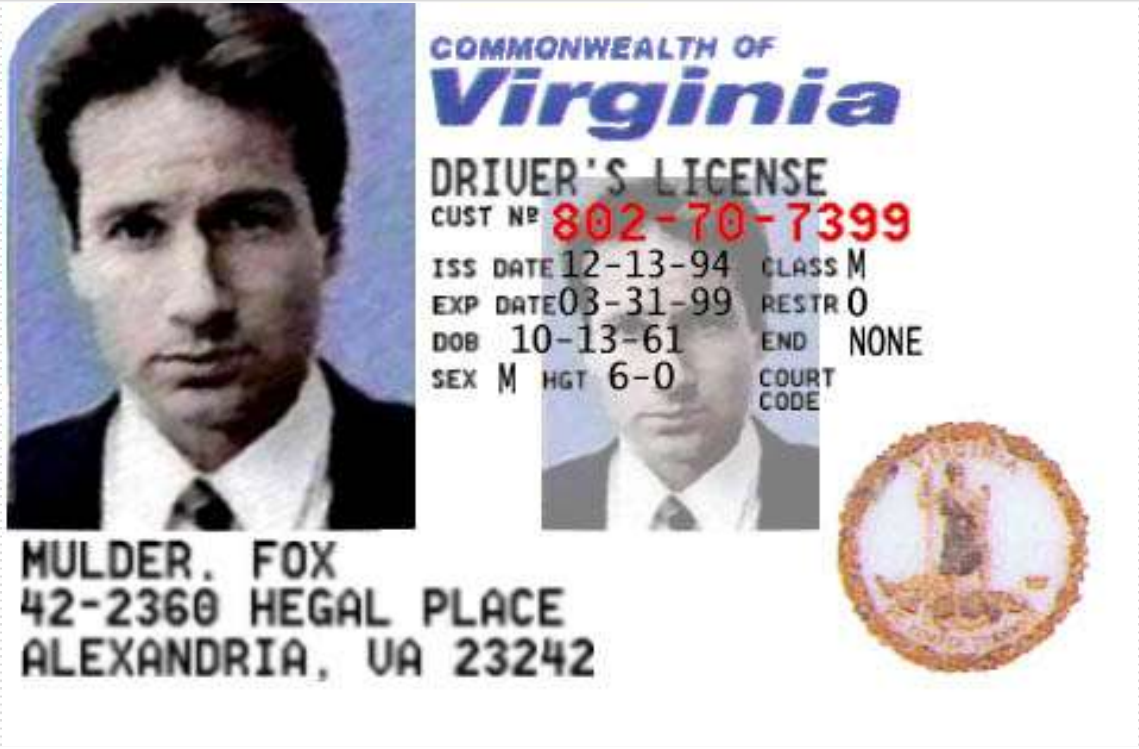
you can always trust a photo id.

you can’t go wrong with photo id

what’s wrong with this picture?

i never sign a key that doesn't have a real name on it. you can always verify a real name with photo id.

the importance of checking photo id



do you recognize this man?

identity verification: a real person

just another average person…

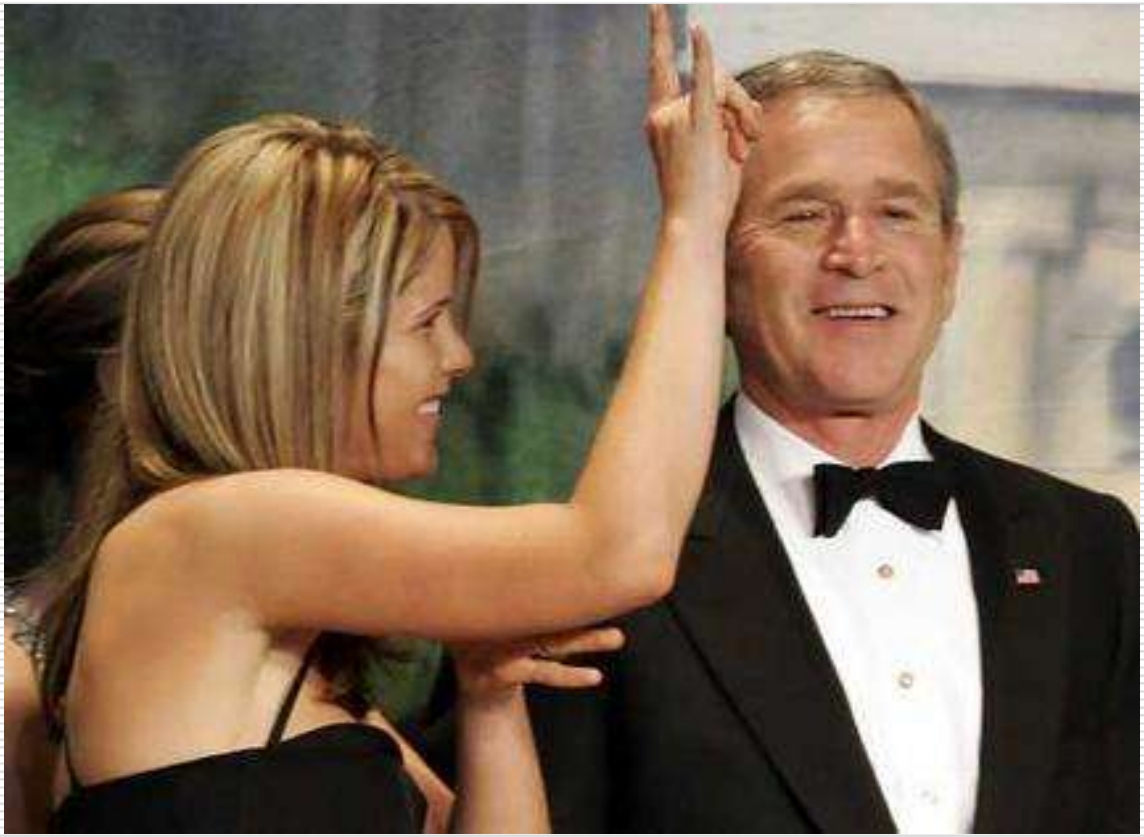
a photo id gives out many details...

her name?
her address?

a photo id gives out many details...

why does she look familiar?

a photo id gives out many details...



here's a picture of her and her father!

iii. non-crypto applications

trusting information

```
pub   1024D/5E345628 2002-01-24
uid                  mailto:shardy@aculei.net
sig 2        AF9929E4 2004-06-16  Justin Brzozoski <jski@gweep.net>
sig 3        DA5BFE1D 2004-07-17  Miles Nordin <carton@ivy.net>
sig 1        1F15AA42 2004-05-25  mangala (Aculei Animi) <mangala@aculei.net>
uid                  mailto:shardy@gmail.com
sig          3D883EA0 2004-12-31  Hendrik Scholz <hscholz@wormulon.net>
sig          42B654AB 2005-01-09  Erik Scharwaechter <diozaka@gmx.de>
sig 3        44030C12 2005-01-01  Andreas Leibrock <fh@leibi.net>
```

- which email address is better?
  - who do you think knows me better?
  - would you trust someone more if they email me more?

trusting information

```
pub   1024D/5E345628 2002-01-24
uid                  phone:+16175551212
sig 2        AF9929E4 2004-06-16  Justin Brzozoski <jski@gweep.net>
sig 3        DA5BFE1D 2004-07-17  Miles Nordin <carton@ivy.net>
sig 1        1F15AA42 2004-05-25  mangala (Aculei Animi) <mangala@aculei.net>
uid                  phone:+15089991212
sig          3D883EA0 2004-12-31  Hendrik Scholz <hscholz@wormulon.net>
sig          42B654AB 2005-01-09  Erik Scharwaechter <diozaka@gmx.de>
sig 3        44030C12 2005-01-01  Andreas Leibrock <fh@leibi.net>
```

which phone number is better?

who do you think knows me better?

can we use existing social networks?

verifying info, asserting trust

what if user ids weren't limited to ones attached to a person's key?

what if a user id had nothing to do with a key?

idea: sign address book data, push it out via FOAF

foaf

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:admin="http://webns.net/mvcb/">
<foaf:Person rdf:nodeID="me">
<foaf:name>seth hardy</foaf:name>
<foaf:title>mr</foaf:title>
<foaf:givenname>seth</foaf:givenname>
<foaf:family_name>hardy</foaf:family_name>
<foaf:nick>shardy</foaf:nick>
<foaf:mbox_sha1sum>69f03f7b91e23ed335a6080ab245a2d6b7840a48
  </foaf:mbox_sha1sum>
<foaf:homepage rdf:resource="http://www.aculei.net/~shardy"/>
<foaf:phone rdf:resource="tel:+1-617-555-1212"/></foaf:Person>
</rdf:RDF>
```

publish foaf info for yourself, others; correlate the data

distributed address book

- what happens if different data given?
- assign trust values to people based on how good they keep information
- leverage power of existing social networks
- problem: trust values may be different from person to person

iv. one last rant

um, excuse me?

from the keysigning party howto:

It's important to note here that some people believe that keeping their public key secret adds an extra degree of security to their encrypted communications. This is true, because a keyserver could be broken or compromised and return the incorrect public key when queried. Further, the key on a given public keyserver may not be the most up to date version of the key. For example, additional signatures may have been added to the key which have not been propagated or uploaded to the keyserver. It is also true because the public key of a key pair is needed to carry out certain types of attacks against the public key cryptosystems which pgp uses. While many people expect, with reasonably large keysizes, that these attacks are so extremely unlikely to be successful that is does not matter if the public key is broadcast, keeping the public key secret does in fact strengthen the key pair.

v. conclusions

how good is ‘good’?

- by current ‘good’ keysigning practice, we can NOT use:
  - pseudonyms
  - organizations
  - informal social networks
- contradictions and blatantly wrong info in ‘official’ documentation
- people refusing to sign keys because of information that is inaccurate

you must trust something

- ultimately you need to trust some link in the system…
- photo ids, other documents can be forged
  - do you ask for a birth certificate?
  - talk to the person's family?
  - social reputations may be more fault tolerant but have no paper
- can you trust anything you can't verify with your own two eyes (e.g. photo uids)?
- why not trust things you know you can trust, instead of what people say you should?

questions?