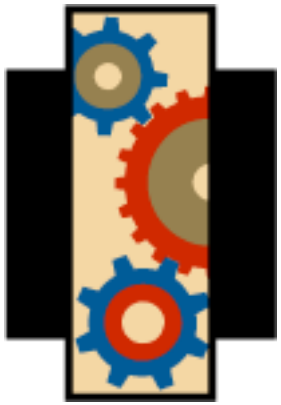


Trust Reflection

A distributed approach to PGP key signing
at multi-day events

Joe Abley, ISC

NANOG 34
Seattle, WA, USA

About PGP

- General-purpose tool for data encryption and non-repudiation
- Most commonly (?) used for e-mail
- Used to sign source code (e.g. ISC, BIND)
- Effective encryption and non-repudiation depends on access to trusted copies of public keys

Web-of-Trust

- You don't need to have personally signed someone else's key in order to be able to trust it
  - but you do need to have laid the groundwork for a web of trust by signing keys, and having your key signed
- Opportunistic use of PGP relies on regular, widespread key-signing

Key Signing Parties

- A method to equip a group of people with a list of known-trustworthy public key fingerprints
- Public keys can be retrieved from untrustworthy sources, and trusted (or not trusted) based on their fingerprints

Fingerprint Verification

- Someone reads out a fingerprint
- The owner of the fingerprint compares what is read aloud with a trusted copy of the fingerprint, and confirms whether it is accurate
- Everybody else follows along with a personal copy of the fingerprint, and annotates accordingly

Identity Verification

- Everybody takes appropriate steps to ensure that the person who just validated the public key fingerprint really is who they purport to be
  - government-issued photo-ID
  - reaction of others in the room
  - whatever suits the individual

Key Signing

- Obtain each public key
- Generate a local fingerprint of the public key
- Compare it with the trusted copy of the fingerprint obtained from the key signing party, annotated with notes on identity verification
- If the fingerprints match, sign the key

E-mail Verification

- Encrypt the exported, signed key towards the key itself
- Mail the encrypted, signed key to the addresses listed in the key's uid
- The signature will only escape into the wild if the mail is able to be received by someone with the corresponding private key

Scaling Problems

- For a key-signing party of n people:
  - fingerprint verification scales linearly, $O(n)$
  - identify verification is quadratic, $O(n^2)$
  - key signing and e-mail verification do not have to happen at the key signing party

Scheduling Issues

- NANOG-Specific problems:
  - late on Monday night
  - overlaps with the NSP-SEC BOF, by virtue of the fact that NSP-SEC is useful and hence always runs late

Other Challenges

- People submit their public keys late
- People show up without a trusted copy of their public key fingerprint
- People don't realise they need to actually generate a public key before any of this makes sense
- People only get one chance to get everything right

Other Approaches

- "Efficient Group Key Signing Method", Len Sassaman
  - eliminates some of the horror of reading hexadecimal digits aloud
  - requires more preparation work on the part of attendees
  - doesn't address the identity verification scaling problem at all

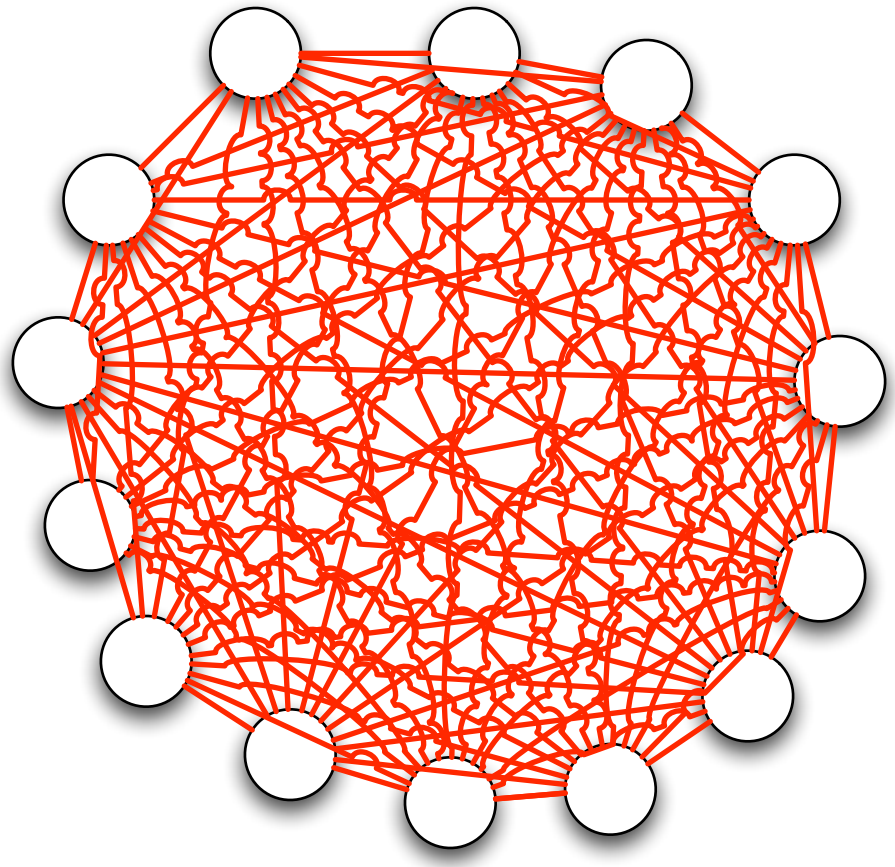
Conclusions

- Big key signing parties are tedious
- Finding time for a big key signing party at NANOG is difficult
- Having more key signing parties in each meeting would give people a better chance to participate

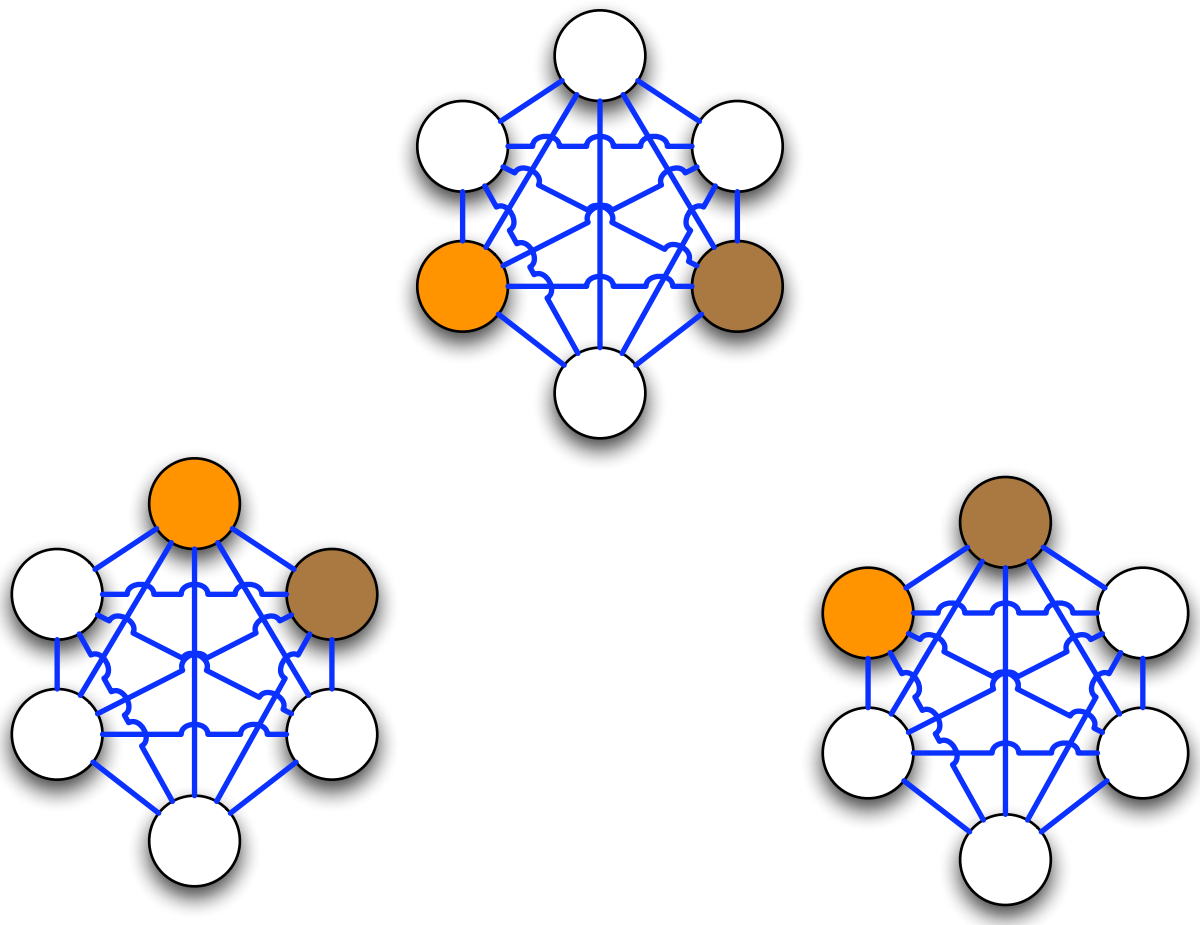
Trust Reflection

- Hold a series of much smaller key parties instead of one big one (e.g. max 8 people)
- Hold each key-signing party in an obvious, accessible and public place
- Rely on volunteers (trust reflectors) to attend all key signing parties, and to act as introducers between attendees at different parties

Hence, this:



becomes this:



NANOG 34

- PGP Key Signing Parties will be held in the last 10-15 minutes of every break, in the Terminal Room
- There will be a daily set of fingerprint sheets printed every morning, so submit your key the day before you plan to attend
- Bring a pen, photo ID and a trusted copy of your public key fingerprint

References

- http://www.nanog.org/pgp.abley.html
- http://www.isc.org/pubs/pres/NANOG/34/trustreflector.pdf
- http://www.cryptnet.net/fdp/crypto/gpg-party.html
- http://sion.quickie.net/keysigning.txt