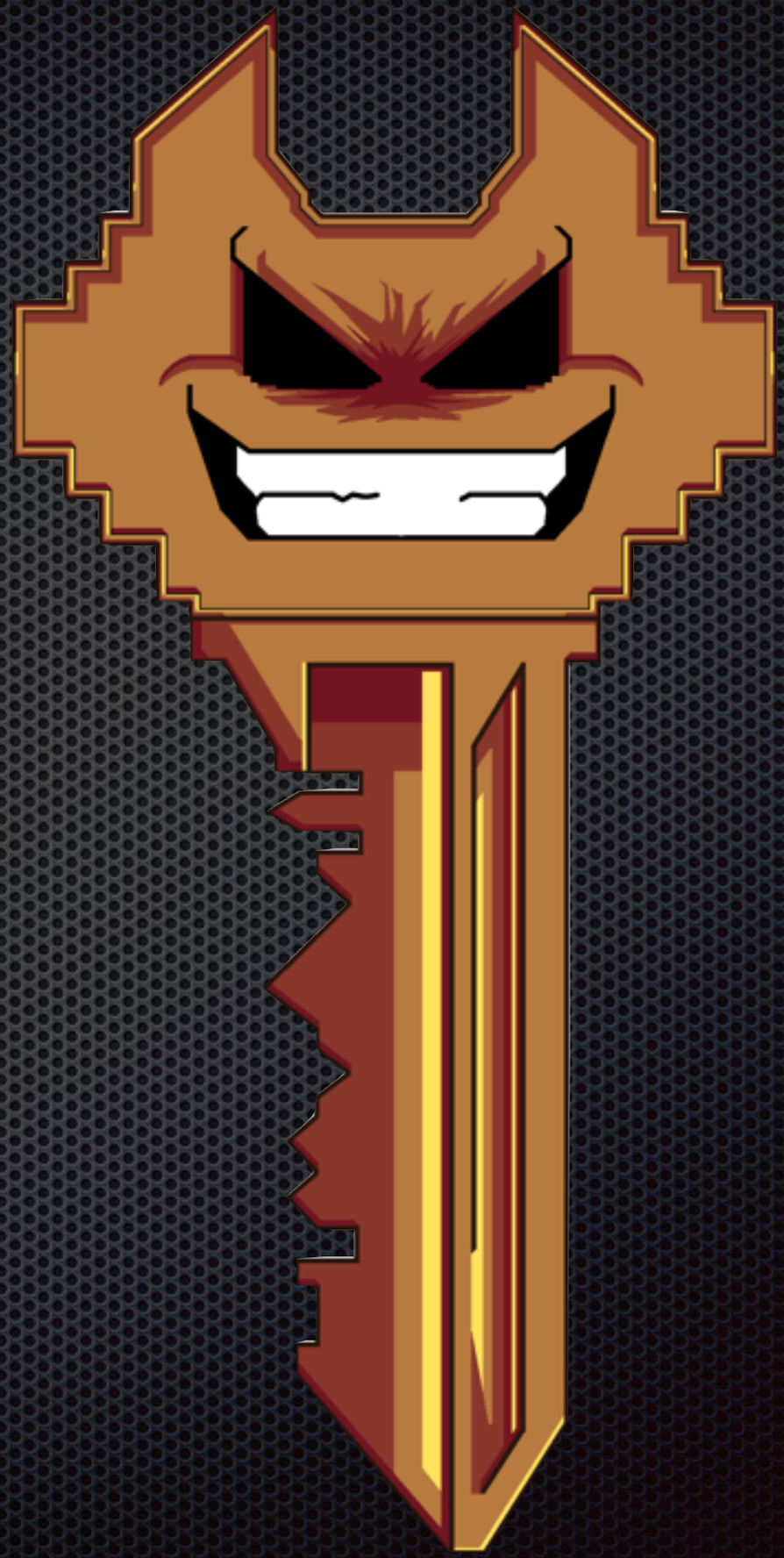
Evil 32

Cause 32 Bit ain’t enough

von Stean



Agenda

- Bedeutung von GPG
- Was sind Key-IDs?
- Was sind Fingerprints?
- Schwachstelle/Angriff
- Demo

Hinweis:
Ich erzähle nur über den Angriff und habe nicht selbst beim Finden der Idee oder der Entwicklung der Software mitgeholfen

Wer hat noch keine mit GPG verschlüsselte Mail verschickt?

GPG

- Das wahrscheinlich weltweit verbreitetste System zur Mailverschlüsselung
- verwendet Asymmetrische Verschlüsselung (private/public Key)
- Ein Haufen Nicht-Cryptonerds verlässt sich auf die Sicherheit von GPG
  - Journalisten
  - Anwälte
  - Aktivisten
  - Software Maintainer & User größerer Linux-Distros
  - unsere Cryptopartybesucher?

Einsatzzwecke von GPG

- Mailverschlüsselung
- Paketsignierung
- Verschlüsselung von Dateien/Backups

Schlüsselaustausch

- Transport des Public-Keys in Form…
  - einer Datei
    - USB-Stick
    - E-Mail
    - …
  - eines Downloads vom Keyserver
    - anhand von Name/ Mailadresse
    - anhand von 32/64-Bit suffix v. Fingerabdruck

Was ist der Fingerabdruck/Key-ID eines öffentlichen GPG-Schlüssels?

- Fingerabdruck = SHA-1 Hash (160 Bit) von Public-Key
  - Bsp: E9D4 691F 0358 15CB 3D46 95BA 9F57 328F 1AB6 5318

Was ist der Fingerabdruck/Key-ID eines öffentlichen GPG-Schlüssels?

- Fingerabdruck = SHA-1 Hash (160 Bit) von Public-Key
  - Bsp: E9D4 691F 0358 15CB 3D46 95BA 9F57 328F 1AB6 5318
- Key-ID = Die letzten 32 oder 64 Bit des Fingerabdrucks
  - Beispiel: 1AB6 5318
  - Vorteil: kürzer zu schreiben
  - Nachteil: zu viele/einfache Kollisionen

Problem?

- Menschen sind sehr schlecht beim Vergleichen von großen Strings
- In manuellen Vergleichen und internen Sicherheitschecks wird manchmal anstelle des kompletten Fingerprints nur die 32-Bit Key-ID verwendet

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar
  - Schlüsselpaar erzeugen, bei dem die letzten 32 Bit des Fingerprints dem des zu fälschenden Keys entsprechen

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar
  - Schlüsselpaar erzeugen, bei dem die letzten 32 Bit des Fingerprints dem des zu fälschenden Keys entsprechen
  - öffentlichen Schlüssel in den Schlüsselbund der Zielperson bringen

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar
  - Schlüsselpaar erzeugen, bei dem die letzten 32 Bit des Fingerprints dem des zu fälschenden Keys entsprechen
    - öffentlichen Schlüssel in den Schlüsselbund der Zielperson bringen
  - Manipulierte Software mit generiertem Schlüssel signieren

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar
  - Schlüsselpaar erzeugen, bei dem die letzten 32 Bit des Fingerprints dem des zu fälschenden Keys entsprechen
    - öffentlichen Schlüssel in den Schlüsselbund der Zielperson bringen
  - Manipulierte Software mit generiertem Schlüssel signieren
  - Bei z.B. MITM-Angriff anstelle des echten Paketes unterschieben

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar
  - Schlüsselpaar erzeugen, bei dem die letzten 32 Bit des Fingerprints dem des zu fälschenden Keys entsprechen
    - öffentlichen Schlüssel in den Schlüsselbund der Zielperson bringen
- Manipulierte Software mit generiertem Schlüssel signieren
- Bei z.B. MITM-Angriff anstelle des echten Paketes unterschieben
- Profit

Angriff

- Szenario: Ein Paketmanager führt die Signaturchecks nicht korrekt durch und ist dadurch verwundbar
  - Schlüsselpaar erzeugen, bei dem die letzten 32 Bit des Fingerprints dem des zu fälschenden Keys entsprechen
    - öffentlichen Schlüssel in den Schlüsselbund der Zielperson bringen
  - Manipulierte Software mit generiertem Schlüssel signieren
  - Bei z.B. MITM-Angriff anstelle des echten Paketes unterschieben
  - Profit

Wie?

Scallion

- Tool, zum Generieren von Schlüsseln, die bestimmte Bedingungen erfüllen
- Vorgehen:

1. RSA-Schlüssel mit libOpenSSL generieren
2. Schlüssel zur GPU senden
3. Exponent erhöhen
4. Schlüssel hashen
5. Wenn partielle Kollision -> Zurück zu Schritt 3
6. Schlüssel zurück zur CPU senden
7. Brandneuer Schlüssel mit partieller Kollision

Demo

Scallion

Scallion

- scallion.exe --gpg --timestamp 12345678 02342CCC$

Scallion

Found new key! Found 1 unique keys.
<XmlMatchOutput>
<GeneratedDate>2008-10-30T11:10:58.7951954Z</GeneratedDate>
<Hash>xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx5eb7cb21</Hash>
<PrivateKey>-----BEGIN PGP PRIVATE KEY BLOCK-----
Version: Scallion

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Scallion

- scallion.exe --gpg --timestamp 12345678 02342CCC$
- Privaten Schlüssel in Datei priv.key speichern

Scallion

- scallion.exe --gpg --timestamp 12345678 02342CCC$
  - Privaten Schlüssel in Datei priv.key speichern
- gpg --import --allow-non-selfsigned-uid priv.key

Search results for '0x6f9ea68a'

https://pgp.mit.edu/pks/lookup?search=0x6F9EA68A&op=index

Search

Search results for '0x6f9ea68a'

```
Type bits/keyID     Date       User ID
```

```
pub  1024D/6F9EA68A 2002-08-09 Lars Kasper <mail@LarsKasper.de>
                               Lars Kasper <Lars.Kasper@gmx.de>
                               Lars Kasper <Lars.Kasper@web.de>
                               Lars Kasper <lars.kasper@gmail.com>
```

Echt.

```
pub  1024R/6F9EA68A 2014-06-16 *** KEY REVOKED *** [not verified]
                               Lars Kasper <mail@LarsKasper.de>
```

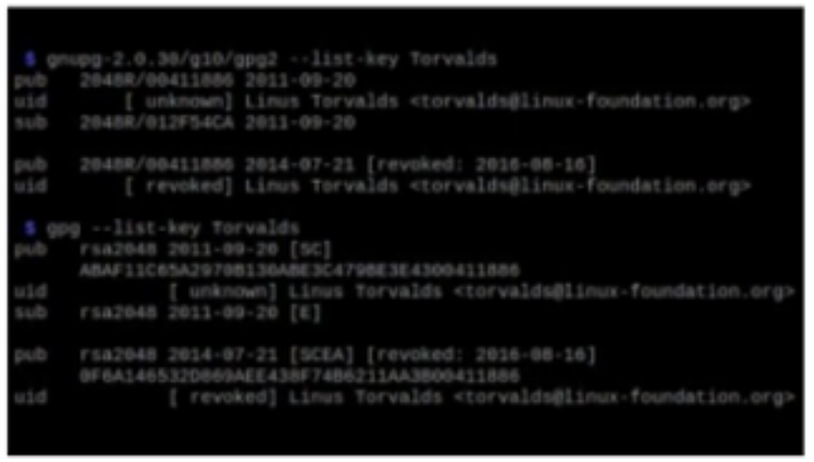
Gefälscht!

KEYSERVER

Chaos mit doppelten PGP-Key-IDs

Auf den PGP-Keyservern sind massenhaft Kopien von existierenden PGP-Keys mit der identischen Key-ID aufgetaucht. Die stammen von einem Experiment von vor zwei Jahren. Key-IDs dürften damit ausgedient haben, als Ersatz sollte man den gesamten Fingerprint nutzen.

"Fake-Keys von Linus Torvalds in freier Wildbahn gefunden" heißt es jüngst in einer Mail an die Linux-Kernel-Mailingliste. Darin wird erklärt, dass sich auf den öffentlichen PGP-Keyservern zwei Keys von Linus Torvalds mit derselben Key-ID - 00411886 - befinden. Doch Torvalds ist nicht der einzige Betroffene, offenbar wurden von unzähligen Keys Duplikate hochgeladen.



Zwei Keys mit derselben Key-ID - in neueren GnuPG-Versionen wird der Fingerprint angezeigt, der eine Unterscheidung ermöglicht. (Bild: Screenshot Hanno Böck)

Datum: 17.8.2016, 09:27

Autor: Hanno Böck

Themen: PGP, Def Con 2014, Defcon, Fingerprinting, GPG, Telekommunikation, Verschlüsselung, Applikationen, Open Source, Security

Schutz

- kompletten Fingerprint anstelle der Key-ID zum überprüfen verwenden
- Web of Trust