old trust-anchors on github

Roy Arends

Paul Hoffman

IEPG102

What?

Warren Kumari:

> Many files on github contain only the old trust anchor.
>
> 2070 files with the old trust-anchor
>
> 412 files with the new trust-anchor

OLD TA: 49aac11d7b6f6446702e54a16…

2070 code results:

- 411 C++
- 384 C
- 280 Python
- 56 XML
- 35 Ruby
- 29 INI
- 29 Shell
- 23 HTML
- 21 Markdown
- 15 Text
- etc

NEW TA: E06d44b80b8f1d39a95c0b0d7…

412 code results:

- 153 C
- 8 Ruby
- 7 Python
- 6 Javascript
- 6 reStructuredText
- 4 HTML
- 4 XML
- 3 JSON
- 2 Clojure
- 2 DNS Zone
- etc

Web Search function on github is limited

- Good thing there is a search API
- Which is limited to 1000 results
  - In buckets of 100
- And rate limited.

- Solution: Use logarithmic file size buckets, in a delayed loop
- Use curl to fetch JSON results
- Use JQ to consult JSON results
- Use grep-sort-uniq-comm for the rest

Files belong to repositories

- Repositories have unique identifiers
- Create two sets:
  - Repositories with ”old-ta” files: 1312
  - Repositories with “new-ta” files: 366
  - “comm -23 old-ta new-ta”

1099 repositories with only the old trust anchor

Not all old-ta files are unique

- In github, files have unique hashes
- From the 1099 repositories with only old-ta listed:

301 unique files

What is the distribution?

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"
195 "430e51a8068a7be23684c7554f8cde4e720d7f64"
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
 56 "81bb896f7170d0ceffecd6147aa02210a59f6a11"
 30 "68e8e405f7149b17d4dcbb372150d1e023783136"
 27 "138e19b51a12d6954f0f05a783f3ae2e1dec513b"
 24 "e1255c4d5e40a458938bd4118e4f8bf1a451295b"
 23 "c48b907ba1275e9fefaf3f6bb40ef68828b337b2"
 20 "9df0d95b417ce6c550a8575ef2ae7c6b1b410bf8”
```

What is the distribution?

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"
195 "430e51a8068a7be23684c7554f8cde4e720d7f64"
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
 56 "81bb896f7170d0ceffecd6147aa02210a59f6a11"
 30 "68e8e405f7149b17d4dcbb372150d1e023783136"
 27 "138e19b51a12d6954f0f05a783f3ae2e1dec513b"
 24 "e1255c4d5e40a458938bd4118e4f8bf1a451295b"
 23 "c48b907ba1275e9fefaf3f6bb40ef68828b337b2"
 20 "9df0d95b417ce6c550a8575ef2ae7c6b1b410bf8”
```

name distribution of the first element:

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"


180 "root.key"
 19 "root.autokey"
 10 "0f4a4ec21ea5b6b392d7a350d10e5d1375982062.svn-base"
  5 "root.key.svn-base"
  3 "root.ds"
  3 "root.autokey.svn-base"
  3 "ksk-as-ds.txt"
  1 "root-anchor"
```

How are these used?

- Lets look at the paths in the repositories:

```
105 "net/unbound/files/root.key"
 20 "feeds/packages/net/unbound/files/root.key"
 14 "feeds/oldpackages/net/unbound/files/root.key"
  8 "net/unbound/files/root.autokey"
  8 "feeds/packages/.svn/pristine/0f/0f4a4ec21ea5b6b392d7a35......
  3 "usr/share/dns/root.ds"
  3 "qsdk/qca/feeds/packages/net/unbound/files/root.key"
  3 "packages/net/unbound/files/root.key"
  3 "openwrt/qca/feeds/packages/net/unbound/files/root.key"
  3 "net/unbound/files/.svn/text-base/root.key.svn-base"
```

How are these used?

- Lets look at the paths in the repositories:

```
105 "net/unbound/files/root.key"
 20 "feeds/packages/net/unbound/files/root.key"
 14 "feeds/oldpackages/net/unbound/files/root.key"
  8 "net/unbound/files/root.autokey"
  8 "feeds/packages/.svn/pristine/0f/0f4a4ec21ea5b6b392d7a35......
  3 "usr/share/dns/root.ds"
  3 "qsdk/qca/feeds/packages/net/unbound/files/root.key"
  3 "packages/net/unbound/files/root.key"
  3 "openwrt/qca/feeds/packages/net/unbound/files/root.key"
  3 "net/unbound/files/.svn/text-base/root.key.svn-base"
```

Why?

- Almost all of these repositories are related to OpenWRT
- Some forks
- Some clones
- Some archived
- All are “config overlay” repositories
  - Install openwrt, and fetch a repository for a specific configuration

- Most of these haven’t been updated since 2015

What is the distribution?

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"
195 "430e51a8068a7be23684c7554f8cde4e720d7f64"
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
 56 "81bb896f7170d0ceffecd6147aa02210a59f6a11"
 30 "68e8e405f7149b17d4dcbb372150d1e023783136"
 27 "138e19b51a12d6954f0f05a783f3ae2e1dec513b"
 24 "e1255c4d5e40a458938bd4118e4f8bf1a451295b"
 23 "c48b907ba1275e9fefaf3f6bb40ef68828b337b2"
 20 "9df0d95b417ce6c550a8575ef2ae7c6b1b410bf8”
```

195 "430e51a8068a7be23684c7554f8cde4e720d7f64"

- All 195 files have the exact same name:

  “dnssec_test.py”

How are these used?

- Lets look at the paths again:

```
103 "external/unbound/libunbound/python/examples/dnssec_test.py"
 44 "contrib/unbound/libunbound/python/examples/dnssec_test.py"
 21 "libunbound/python/examples/dnssec_test.py"
  6 "Modules/unbound/libunbound/python/examples/dnssec_test.py"
  5 "usr/src/contrib/unbound/libunbound/python/examples/dnssec_test.py"
  5 "external/bsd/unbound/dist/libunbound/python/examples/dnssec_test.py"
  1 "stellite/external/unbound/libunbound/python/examples/dnssec_test.py"
  1 "src/contrib/monero-0.10.0/external/unbound/libunbound/python/ex…
  1 "shangcoin/external/unbound/libunbound/python/examples/dnssec_test…
  1 "moneda/external/unbound/libunbound/python/examples/dnssec_test.py"
```

How are these used?

- Lets look at the paths again:

```
103 "external/unbound/libunbound/python/examples/dnssec_test.py"
 44 "contrib/unbound/libunbound/python/examples/dnssec_test.py"
 21 "libunbound/python/examples/dnssec_test.py"
  6 "Modules/unbound/libunbound/python/examples/dnssec_test.py"
  5 "usr/src/contrib/unbound/libunbound/python/examples/dnssec_test.py"
  5 "external/bsd/unbound/dist/libunbound/python/examples/dnssec_test.py"
  1 "stellite/external/unbound/libunbound/python/examples/dnssec_test.py"
  1 "src/contrib/monero-0.10.0/external/unbound/libunbound/python/ex…
  1 "shangcoin/external/unbound/libunbound/python/examples/dnssec_test…
  1 "moneda/external/unbound/libunbound/python/examples/dnssec_test.py"
```

Why?

- Lets look at the paths again again:

```
103 "external/unbound/libunbound/python/examples/dnssec_test.py"
 44 "contrib/unbound/libunbound/python/examples/dnssec_test.py"
 21 "libunbound/python/examples/dnssec_test.py"
  6 "Modules/unbound/libunbound/python/examples/dnssec_test.py"
  5 "usr/src/contrib/unbound/libunbound/python/examples/dnssec_test.py"
  5 "external/bsd/unbound/dist/libunbound/python/examples/dnssec_test.py"
  1 "stellite/external/unbound/libunbound/python/examples/dnssec_test.py"
  1 "src/contrib/monero-0.10.0/external/unbound/libunbound/python/ex…
  1 "shangcoin/external/unbound/libunbound/python/examples/dnssec_test…
  1 "moneda/external/unbound/libunbound/python/examples/dnssec_test.py"
```

What is moneda/stellite/monero/shangcoin?

- All are crypto currencies
- All are a clone, copy or fork from “monero”
- Example code from the unbound/smallapp/libunbound path
- This is not actively used in these packages

What is the distribution?

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"
195 "430e51a8068a7be23684c7554f8cde4e720d7f64"
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
 56 "81bb896f7170d0ceffecd6147aa02210a59f6a11"
 30 "68e8e405f7149b17d4dcbb372150d1e023783136"
 27 "138e19b51a12d6954f0f05a783f3ae2e1dec513b"
 24 "e1255c4d5e40a458938bd4118e4f8bf1a451295b"
 23 "c48b907ba1275e9fefaf3f6bb40ef68828b337b2"
 20 "9df0d95b417ce6c550a8575ef2ae7c6b1b410bf8”
```

```
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
```

- All 181 files have the exact same name:

  "trust-anchors.conf"

How are these used?

- Lets look at the paths again:

```
 47 "trunk/user/dnsmasq/dnsmasq-2.7x/trust-anchors.conf"
 46 "trust-anchors.conf"
 40 "release/src/router/dnsmasq/trust-anchors.conf"
 11 "release/src-rt-6.x.4708/router/dnsmasq/trust-anchors.conf"
  7 "usr/share/dnsmasq-base/trust-anchors.conf"
  3 "squashfs-root/usr/share/dnsmasq/trust-anchors.conf"
  3 "asuswrt/release/src/router/dnsmasq/trust-anchors.conf"
  2 "user/dnsmasq/trust-anchors.conf"
  2 "rootfs/dnsmasq-2.72/trust-anchors.conf"
  2 "dnsmasq/trust-anchors.conf"
```

How are these used?

- Lets look at the paths again:

```
 47 "trunk/user/dnsmasq/dnsmasq-2.7x/trust-anchors.conf"
 46 "trust-anchors.conf"
 40 "release/src/router/dnsmasq/trust-anchors.conf"
 11 "release/src-rt-6.x.4708/router/dnsmasq/trust-anchors.conf"
  7 "usr/share/dnsmasq-base/trust-anchors.conf"
  3 "squashfs-root/usr/share/dnsmasq/trust-anchors.conf"
  3 "asuswrt/release/src/router/dnsmasq/trust-anchors.conf"
  2 "user/dnsmasq/trust-anchors.conf"
  2 "rootfs/dnsmasq-2.72/trust-anchors.conf"
  2 "dnsmasq/trust-anchors.conf"
```

How is this used?

- It is not used at all as a default configuration
- All dnsmasq related repositories have dnssec off by default
- This can be seen in the configuration files
  - (which are all the same as well)
- Most of these repositories are for the ASUS rt-n56u

What is the distribution?

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"
195 "430e51a8068a7be23684c7554f8cde4e720d7f64"
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
 56 "81bb896f7170d0ceffecd6147aa02210a59f6a11"
 30 "68e8e405f7149b17d4dcbb372150d1e023783136"
 27 "138e19b51a12d6954f0f05a783f3ae2e1dec513b"
 24 "e1255c4d5e40a458938bd4118e4f8bf1a451295b"
 23 "c48b907ba1275e9fefaf3f6bb40ef68828b337b2"
 20 "9df0d95b417ce6c550a8575ef2ae7c6b1b410bf8”
```

```
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
```

- All 177 files have the exact same name:

  "dnssec_chain_verifier.cc"

How are these used?

- Lets look at the paths again:

```
98 "external/chromium/net/base/dnssec_chain_verifier.cc"
61 "net/base/dnssec_chain_verifier.cc"
 6 "chromium/net/base/dnssec_chain_verifier.cc"
 3 "platform/external/chromium/net/base/dnssec_chain_verifier.cc"
 3 "android/external/chromium/net/base/dnssec_chain_verifier.cc"
 1 "lge/external/webkit/htmlwebkit/Source/ThirdParty/chromium/net/b
 1 "kk-4.x/external/chromium/net/base/dnssec_chain_verifier.cc"
 1 "android/vendor/lge/external/webkit/htmlwebkit/Source/ThirdParty
 1 "SM-T210_JB_Opensource_SENSE/Platform/external/chromium/net/base
 1 "Platform/external/chromium/net/base/dnssec_chain_verifier.cc"
```

How are these used?

- Lets look at the paths again:

```
98 "external/chromium/net/base/dnssec_chain_verifier.cc"
61 "net/base/dnssec_chain_verifier.cc"
 6 "chromium/net/base/dnssec_chain_verifier.cc"
 3 "platform/external/chromium/net/base/dnssec_chain_verifier.cc"
 3 "android/external/chromium/net/base/dnssec_chain_verifier.cc"
 1 "lge/external/webkit/htmlwebkit/Source/ThirdParty/chromium/net/b
 1 "kk-4.x/external/chromium/net/base/dnssec_chain_verifier.cc"
 1 "android/vendor/lge/external/webkit/htmlwebkit/Source/ThirdParty
 1 "SM-T210_JB_Opensource_SENSE/Platform/external/chromium/net/base
 1 "Platform/external/chromium/net/base/dnssec_chain_verifier.cc"
```

How is this used?

- This is chromium
- More specifically, this code was used for DNSSEC-stapled-certificates
- This has been removed from chromium about 6 years ago

  https://codereview.chromium.org/11293234/

Issue 11293234: Remove support for DNSSEC stapled certificates. (Closed)

Created:
5 years, 8 months ago by agl

Modified:
5 years, 8 months ago

Description

```
Remove support for DNSSEC stapled certificates.

BUG=none
```

What is the distribution?

```
224 "2ad6641345273ada29b584f5fcdd061c3e99e207"
195 "430e51a8068a7be23684c7554f8cde4e720d7f64"
181 "afda5186ee4071eaeac70efb2245a9d25405341b"
177 "e3eeee78c548c43e1db07fa78ed2df113157242e"
 56 "81bb896f7170d0ceffecd6147aa02210a59f6a11"
 30 "68e8e405f7149b17d4dcbb372150d1e023783136"
 27 "138e19b51a12d6954f0f05a783f3ae2e1dec513b"
 24 "e1255c4d5e40a458938bd4118e4f8bf1a451295b"
 23 "c48b907ba1275e9fefaf3f6bb40ef68828b337b2"
 20 "9df0d95b417ce6c550a8575ef2ae7c6b1b410bf8”
```

```
56 "81bb896f7170d0ceffecd6147aa02210a59f6a11”
30 "68e8e405f7149b17d4dcbb372150d1e023783136"
```

- All 86 files have the exact same name:

  "unbound-anchor.c”

- This is the unbound-anchor code, which sole purpose is to update the trust anchor!

Top 10 names used:

```
242 "dnssec_test.py"
203 "trust-anchors.conf"
191 "dnssec_chain_verifier.cc"
185 "root.key"
157 "unbound-anchor.c"
157 "dns_utils.cpp"
 43 "DNSSECSupport.c"
 30 "dnsval.conf"
 28 "defaults.properties"
 20 "dnssec.rb"
```

Top 10 names used:

```
242 "dnssec_test.py"
203 "trust-anchors.conf"
191 "dnssec_chain_verifier.cc"
185 "root.key"
157 "unbound-anchor.c"
157 "dns_utils.cpp"
 43 "DNSSECSupport.c"
 30 "dnsval.conf"
 28 "defaults.properties"
 20 "dnssec.rb"
```

Top 10 names used:

157 "dns_utils.cpp”

Copied functions from unbound-anchor.c

Exclusively used for Monero related repositories

43 "DNSSECSupport.c”

Part of mDNSResponder (bonjour), used as fallback.

Contains code to configure the new trust-anchor via ICANN certificate.

Top 10 names used

30 "dnsval.conf”

Configuration file for DNSSEC-Tools.

Main repository hasn’t been updated since 2015.

28 "defaults.properties”

Part of jitsi, popular video, voice conferencing and chat tool.

DNSSEC is on by default, and only has the old trust-anchor.

In the process of contacting them.

Top 10 names used

20 "dnssec.rb”

DNSRUBY and clones.

I've pinged Alex Dalitz, who maintains the main repo.

How fresh are these repositories?

Count	Last-pushed
304	2018
221	2017
167	2016
207	2015
95	2014
66	2013
34	2012
5	2011

Contacted/next steps

- dnsruby’s Alex Dalitz
- Kirei
- Knotdns folks
  - Old trust-anchor was used in a closed test. No danger to the public
- Unbound folks
  - Unbound-anchor can update itself. No danger to the public
  - Examples are just that, examples
  - Already updated main repository
- Will investigate further and will contact
- More hosting facilities to follow:
  - Gitlab/bitbucket/assembla/sourceforge/Launchpad

Conclusion

- Github has many ~~dead bodies~~ stale copies of some form of unbound
  - Will contact freshest/most-used
- Many router firmware repositories
  - Openwrt, rt-n56u, dnsmasq
  - In default config, no danger to the public
  - Will contact freshest/most-used
- Many bitcoin variants
- Most of these are not vulnerable during the keyroll