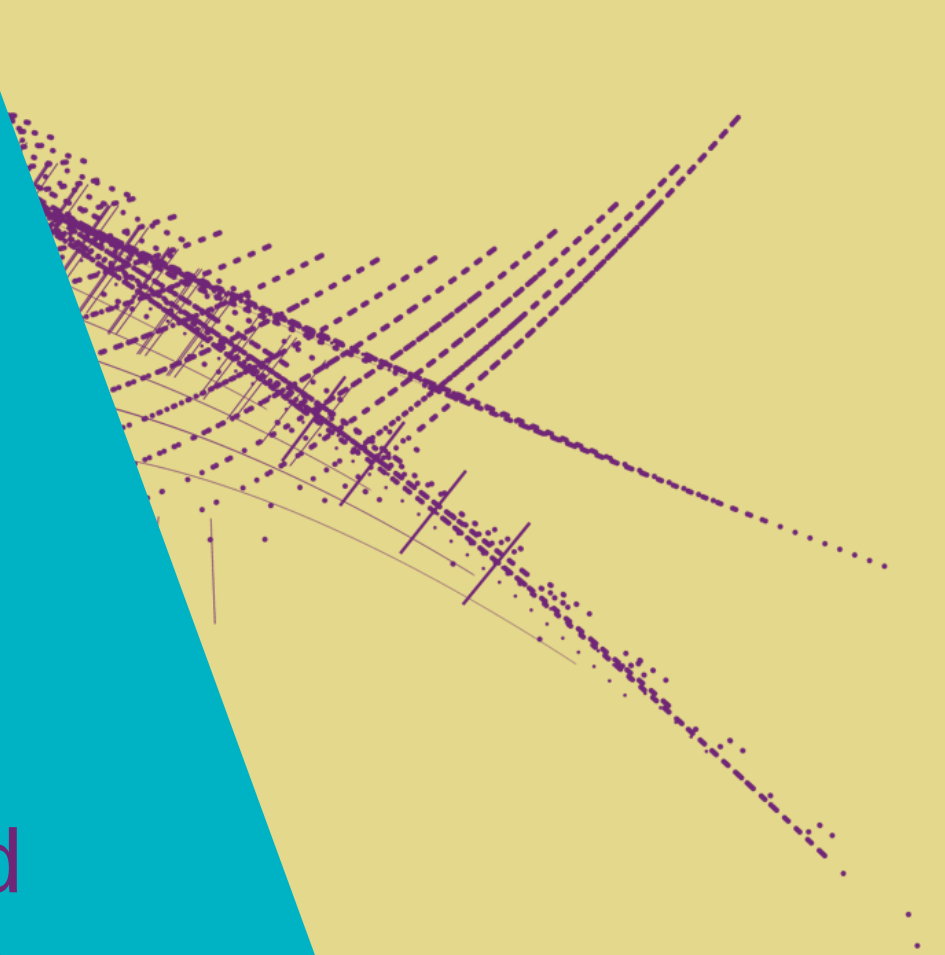
Nikhef

Maastricht University

David Groep

Update: RedHat 9+ and the issue of sha-1 roots



October 2023

Although it conceptually makes no sense …

- We *know* SHA-1 is no longer secure as a crypto digest
  - all EECs and ICAs moved away ages ago
- but some projects and distros are *uselessly* deprecating SHA-1 *for self-signed (root) certificates – where the signature is immaterial*

- This affects at least
  - FF103+
  - RHEL9+ (and rebuilds)
- yet … in the cases we could find it *only* applies to CA certs that are *not* in the WebPKI (and distro) public trust list

This impacts both joint-trust and igtf-only trust when installed in a non-system location. But thy system locations are different is not obvious from the doc …

Rocky9+, AlmaLinux9+, RHEL9+ and

With RHEL9 also deprecating SHA-1, but *at the same time*
still having self-signed SHA-1 based root certs in the ca-certificates package,

unknown why that distribution treats SHA-1 certs in the X509_CERT_DIR differently

At least there is a policy override for now

```
update-crypto-policies --set DEFAULT:SHA1
```

even if that is a rather course-grained tool (and do not pick LEGACY)

The ca-certificates package in RH9

Interestingly, EL9 *does* ship with a lot of SHA-1 root CAs in
ca-certificates-2022.2.54-90.2.el9.noarch.rpm
and the p11-kit sources thereof (and thus e.g. /etc/pki/tls/certs/ca-bundle.crt) contain SHA-1 self-signed roots that do work on EL9.

- this relies on the OSSL proprietary ‘trust bytes’ in a BEGIN TRUSTED CERTIFICATE blob
- such blobs allow SHA-1 for self-signed roots, but are not standarised

Yet the ‘simple’ solution, to ship both the EL/OSSL proprietary ‘trust’ bytes as well as a regular PEM formatted root does *not* work (thanks to Brian Lin for testing that!)

The OSG experiment

OSG shipped the dual-blob mode for a few days

- using something like https://www.nikhef.nl/~davidg/tmp/make-trusted.sh
- first a “BEGIN TRUSTED CERTIFICATE”,
  then *in the same file* “BEGIN CERTIFICATE”

However, it broke:

- CANL-Java, extending BouncyCastle, cannot process this blob and will balk even if it does not recognise it
  (https://stackoverflow.com/questions/55550299/java-can-not-load-begin-trusted-certificate-format-certificate)
- open as a dCache Feature Enhancement on CANL Java by Paul Millar

will not be fixed overnight, of course. And we my find other issues thereafter

Mitigations?

Meanwhile,

- if you still have a SHA-1 root
- and you are able to re-issue with the same key (and new serial)
- and your EECs *do not* have dirname+serial in their AKI

your CAs should probably re-issuing its root because that is easier.

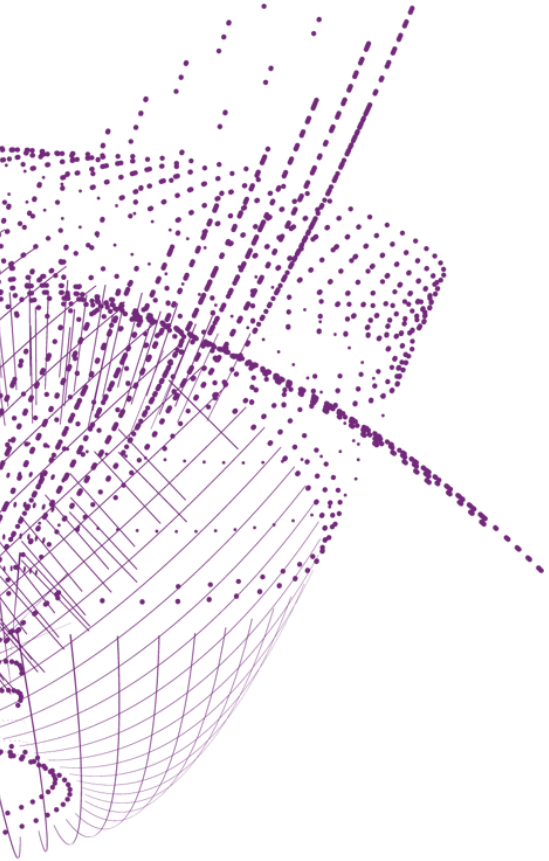
But for the large ones, esp. the DigiCert Assured ID Root from 2006 for instance, that will be hard.
And migrating to another (SHA-2 rooted) signing hierarchy will take at least 395 days ... and a lot of engineering on the RP and CA side

The root cause is with RH not understanding what a self-signed trust anchor is, but that will not help us in the short term.

Reissuance of roots?

ASGCCA-2007
BYGCA
CNIC
DZeScience
DigiCertGridRootCA-Root
KEK
MARGI
RDIG
SRCE
TRGrid
ArmeSFo
CESNET-CA-Root
DFN-GridGermany-Root
DigiCertAssuredIDRootCA-Root
IHEP-2013
LIPCA
QuoVadis-Root-CA2
RomanianGRID
SiGNET-CA
seegrid-ca-2013

Fixed by now: GridCanada, CILogon basic/silver/OpenID, UKeScienceRoot-2007
Removed: DigiCertGridCA-1-Classic, DigiCertGridTrustCA-Classic
Will be discontinued soon: GermanGrid (GridKA)



Discussion

what can *you* do?

David Groep
davidg@nikhef.nl
https://www.nikhef.nl/~davidg/presentations/
https://orcid.org/0000-0003-1026-6606

Maastricht University

Nikhef