IGTF Fabric Updates

status of our authorities, trust fabric news, and RHEL9's (OpenSSL) hash function issues

David Groep
davidg@nikhef.nl

part of the work programme of GEANT 5-1 EnCo, and AARC TREE

the work has received co-funding from the European Union

co-supported by Nikhef and the Dutch National e-Infrastructure coordinated by SURF

Meanwhile in the EUGridPMA+ …

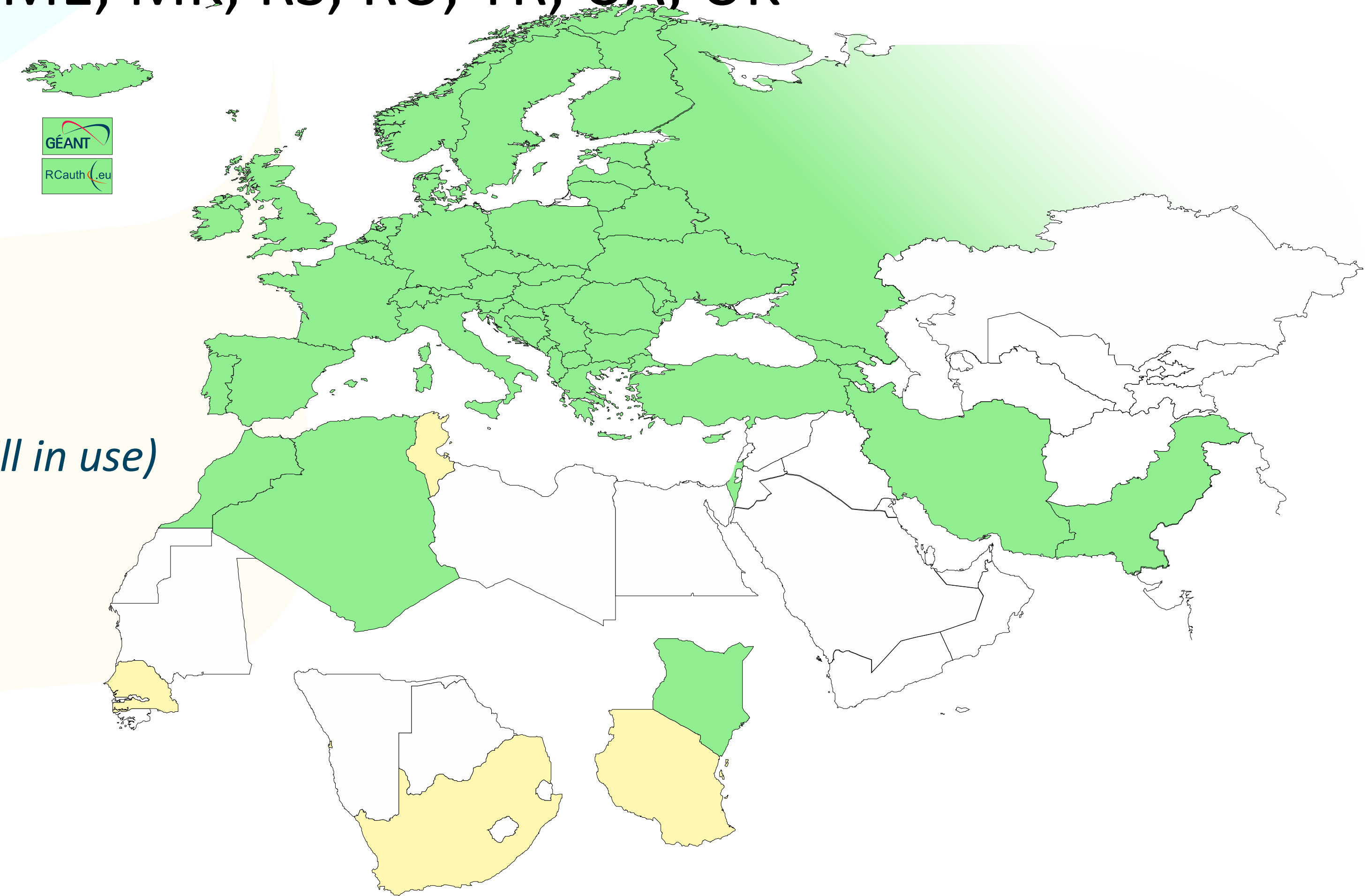
- EUGridPMA and IGTF distribution matters
  - constituency and developments
  - GPG Package Signing Key updates
- S/MIME baseline in CABF: separating authentication and email in TCS
- Root migration update for EL9+ (or: why people bother the fetch-crl devs)

EMEA area membership evolution

- Europe$^+$: GEANT TCS, and CZ, DE, DK(+FI+IS+NO+SE), FR, GR, HR, HU, NL, PL, RO, SI, SK; AM, GE, MD, ME, MK, RS, RU, TR, UA, UK
- Middle East: IR, PK
- Africa: DZ, KE, MA
- CERN, RCauth.eu

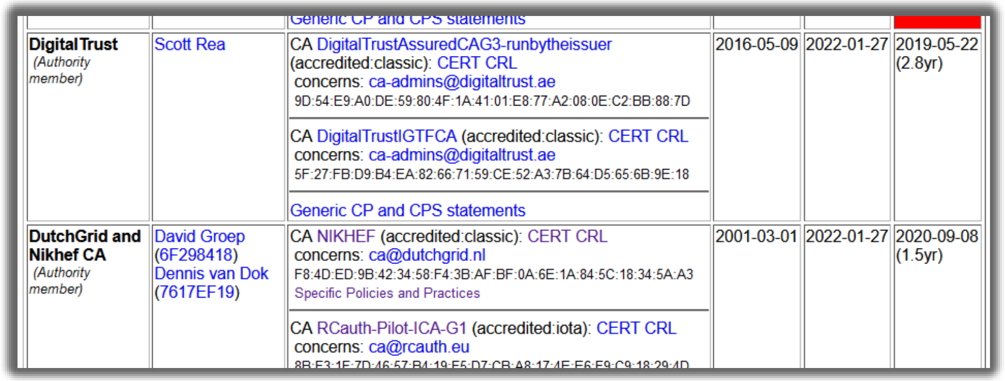
the Swiss moved to eMudhra

(but legacy DutchGrid transitional service still in use)



Membership and other changes

- Identity providers: both reduction and growth
  - migration to GEANT TCS continues: +DE
    https://wiki.geant.org/display/TCSNT/TCS+Participants+Sectigo
  - CERN joining TCS via Renater (FR)
  - Discontinued: -PT, -DE, -QV, -RS
  - Suspended: -KE
- Self-audit review
  - Cosmin Nistor will update us in a moment
  - real-time interaction between authority and reviewers helps, but …
- Issues remain for now with support in .ch – but progress continues!



		Generic CP and CPS statements			🟥
DigitalTrust *(Authority member)*	Scott Rea	CA DigitalTrustAssuredCAG3-runbytheissuer (accredited:classic): CERT CRL concerns: ca-admins@digitaltrust.ae 9D:54:E9:A0:DE:59:80:4F:1A:41:01:E8:77:A2:08:0E:C2:BB:88:7D CA DigitalTrustIGTFCA (accredited:classic): CERT CRL concerns: ca-admins@digitaltrust.ae 5F:27:FB:D9:B4:EA:82:66:71:59:CE:52:A3:7B:64:D5:65:6B:9E:18 Generic CP and CPS statements	2016-05-09	2022-01-27	2019-05-22 (2.8yr)
DutchGrid and Nikhef CA *(Authority member)*	David Groep (6F298418) Dennis van Dok (7617EF19)	CA NIKHEF (accredited:classic): CERT CRL concerns: ca@dutchgrid.nl F8:4D:ED:9B:42:34:58:F4:3B:AF:BF:0A:6E:1A:84:5C:18:34:5A:A3 Specific Policies and Practices CA RCauth-Pilot-ICA-G1 (accredited:iota): CERT CRL concerns: ca@rcauth.eu 8B:F3:1F:7D:46:57:B4:19:F5:D7:CB:A8:17:4F:F6:F9:C9:18:29:4D	2001-03-01	2022-01-27	2020-09-08 (1.5yr)

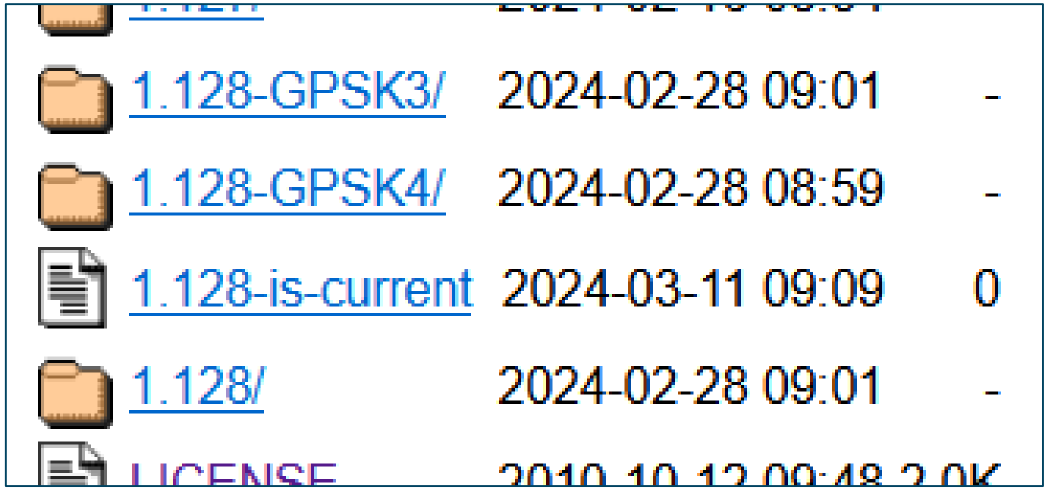
Distribution signing key update

```
error: Verifying a signature using certificate
 D12E922822BE64D50146188BC32D99C83CDBBC71
 (EUGridPMA Distribution Signing Key 3 <info@eugridpma.org>):
 Key C32D99C83CDBBC71 invalid: not signing capable
```

In Fedora Core 38+ (and thus later in its derivatives, and maybe soon in Debian), RSA 1024 package signing no longer supported by default

(work-around with bespoke crypto-policies possible, not recommended)

Distribution signing key update



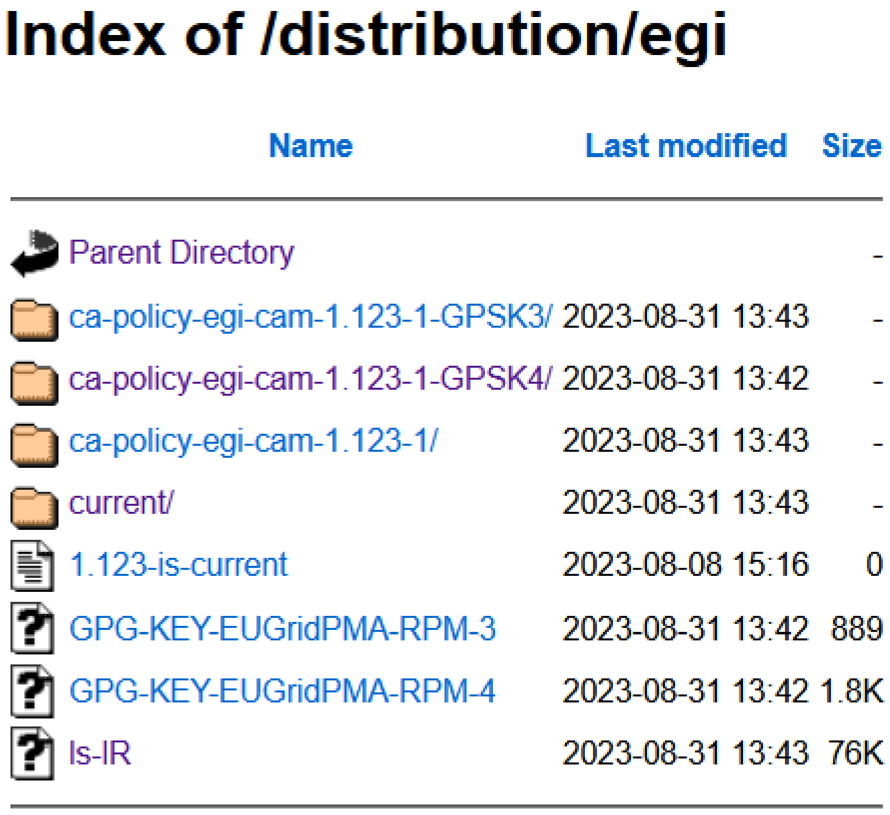
```
1.128-GPSK3/    2024-02-28 09:01   -
1.128-GPSK4/    2024-02-28 08:59   -
1.128-is-current 2024-03-11 09:09  0
1.128/          2024-02-28 09:01   -
LICENSE         2010-10-12 09:48 2.0K
```

In future releases we move to a **new GPG package key**

- RSA-2048
- called GPG-KEY-EUGridPMA-RPM-4
- distributed with 1.122+ releases
- Retrieve new public key file from https://dl.igtf.net/distribution/GPG-KEY-EUGridPMA-RPM-4
- or from the public key servers: rsa/2048 dated 2023-07-29T12:06:23Z
- fingerprint: 565f 4528 ead3 f537 27b5 a2e9 b055 0056 **7634 1f1a**

Specific downstream distribution (like EGI) follow

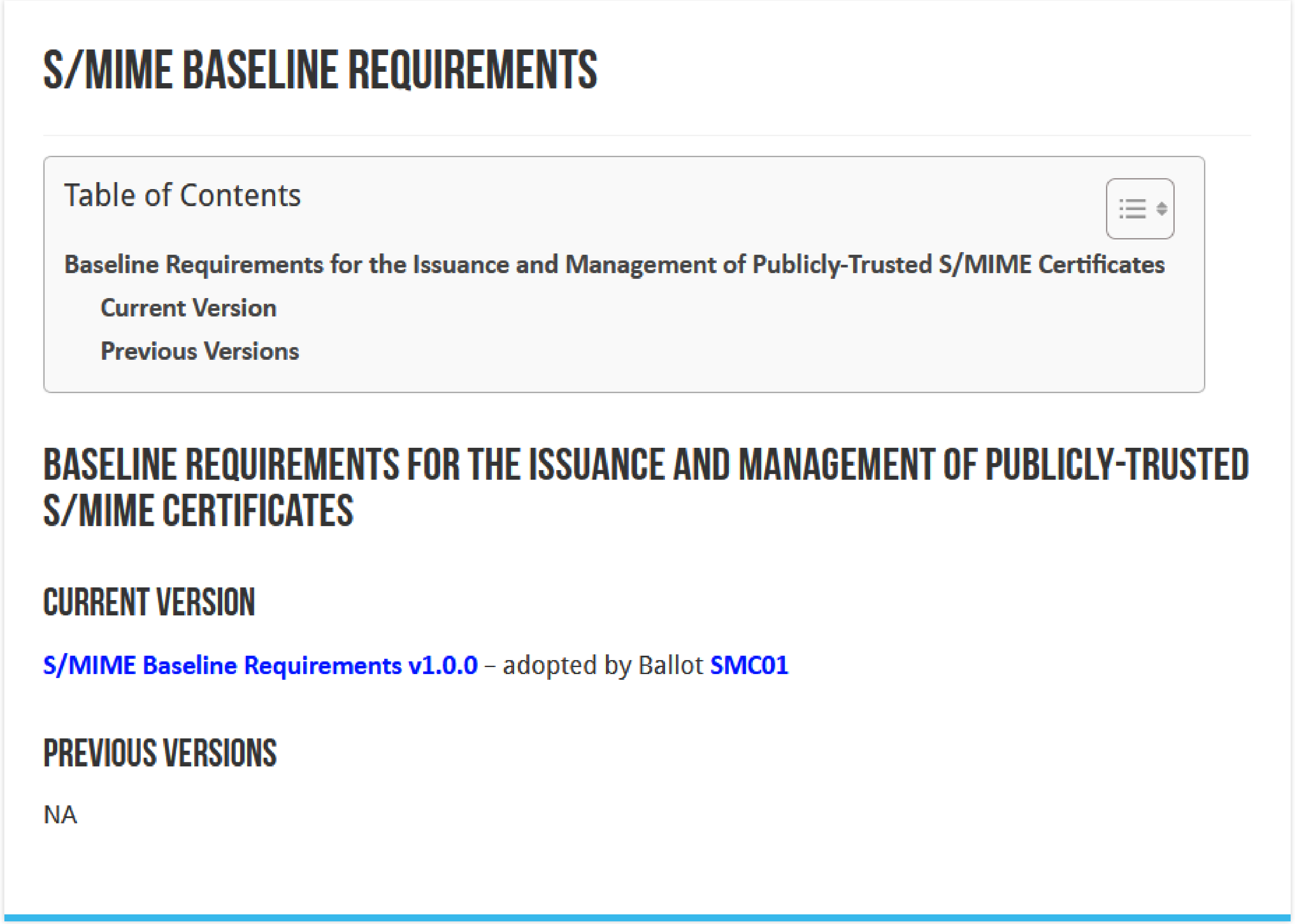
- EGI uses the same signing key, since – for now – the packaging is integrated and co-supported by EGI
- Plan is to move on the next major change, but not before Q2 2024
- RHEL SHA-1 Root issue may be a good time to also make this change the default?



Index of /distribution/egi

Name	Last modified	Size
Parent Directory		-
ca-policy-egi-cam-1.123-1-GPSK3/	2023-08-31 13:43	-
ca-policy-egi-cam-1.123-1-GPSK4/	2023-08-31 13:42	-
ca-policy-egi-cam-1.123-1/	2023-08-31 13:43	-
current/	2023-08-31 13:43	-
1.123-is-current	2023-08-08 15:16	0
GPG-KEY-EUGridPMA-RPM-3	2023-08-31 13:42	889
GPG-KEY-EUGridPMA-RPM-4	2023-08-31 13:42	1.8K
ls-lR	2023-08-31 13:43	76K

CA/BROWSER Forum



S/MIME BASELINE REQUIREMENTS

Table of Contents

BASELINE REQUIREMENTS FOR THE ISSUANCE AND MANAGEMENT OF PUBLICLY-TRUSTED S/MIME CERTIFICATES

CURRENT VERSION

S/MIME Baseline Requirements v1.0.0 – adopted by Ballot SMC01

PREVIOUS VERSIONS

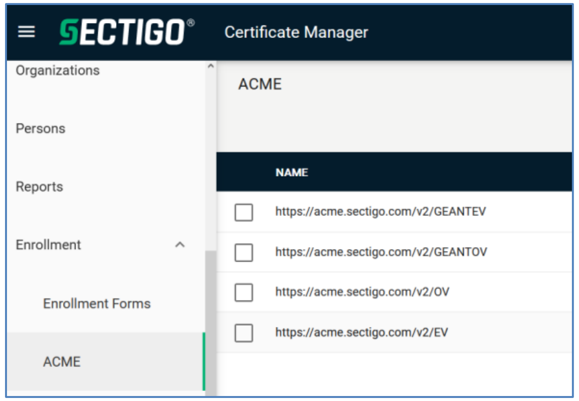
NA

User awareness

- This is a change in communications and documentation as well, not only a set of technical changes
- In request systems, have to clearly distinguish for users *which product to order*. For example:
  - “Personal” stays the same, but is called now “Email signing and Encryption”
  - renaming “IGTF MICS Personal” to “Personal Authentication” and explain
  - renaming “IGTF MICS Robot Personal” to “Personal Automated Authentication”
  - forking “IGTF Classic Robot Email”
    - Authentication-only (IGTF) profile “Classic Robot Email”
    - Email signing profile “Organisation-validated S/MIME signing” (i.e. team-based or role-based)

Other CABF things to keep in mind

- Server SSL BR has already been updated
  - the provision for using DC prefixing has been retained



	NAME
□	https://acme.sectigo.com/v2/GEANTEV
□	https://acme.sectigo.com/v2/GEANTOV
□	https://acme.sectigo.com/v2/OV
□	https://acme.sectigo.com/v2/EV

- But expect shorter validity periods in the future
  - start preparing for 90-day max in your service deployment automation systems
  - increased use of automation (ACME OV using client ID+secret)

```
[root@hekel ~]# certbot certonly \
  --standalone --non-interactive --agree-tos --email davidg@nikhef.nl \
  --server https://acme.sectigo.com/v2/GEANTOV \
  --eab-kid DUniqueID_forthisclient --eab-hmac-key mv_v3ryl0n9s3cr3tK3y \
  --domain hekel.nikhef.nl --cert-name OVGEANTcert
```

THE CHALLENGE OF SELF-SIGNED ROOTS

AND FF & REDHAT’ S IDEA OF WHAT SELF-SIGNED MEANS …

Although it conceptually makes no sense …

- We know SHA-1 is no longer secure – and all EECs and ICAs moved away – when used as a secure hash algorithm. But …
- now, some projects and distros are (uselessly!) deprecating SHA-1 *also for self-signed (root) certificates*

- This affects at least
  - FF103+
  - RHEL9+ (and rebuilds)
- yet … in the cases we could find *only* for 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, depends on a RedHat/OSSL proprietary set of ‘bonus bits’ appended to the end of the ASN.1 certificate blob.

For the others, there is – for now – a policy override:

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

even if that is a rather course-grained and blunt tool

The ca-certificates package in RH9

Interestingly, EL9 *does* ship with a lot of SHA-1 root CAs:

- 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

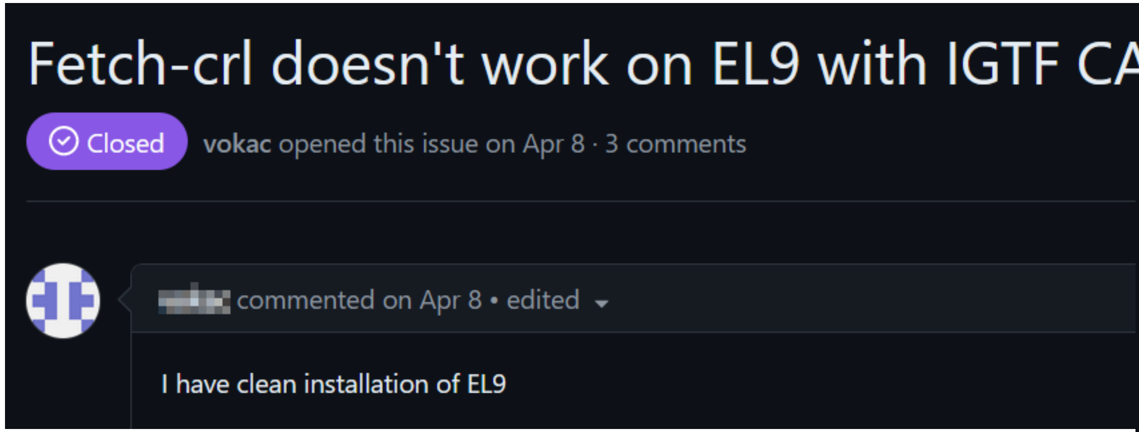
- using something like https://www.nikhef.nl/~davidg/tmp/make-trusted.sh
- first a “BEGIN TRUSTED CERTIFICATE”,
  then *in the same file* “BEGIN CERTIFICATE”

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

End-users don’t understand & open bugs on ‘random’ devs

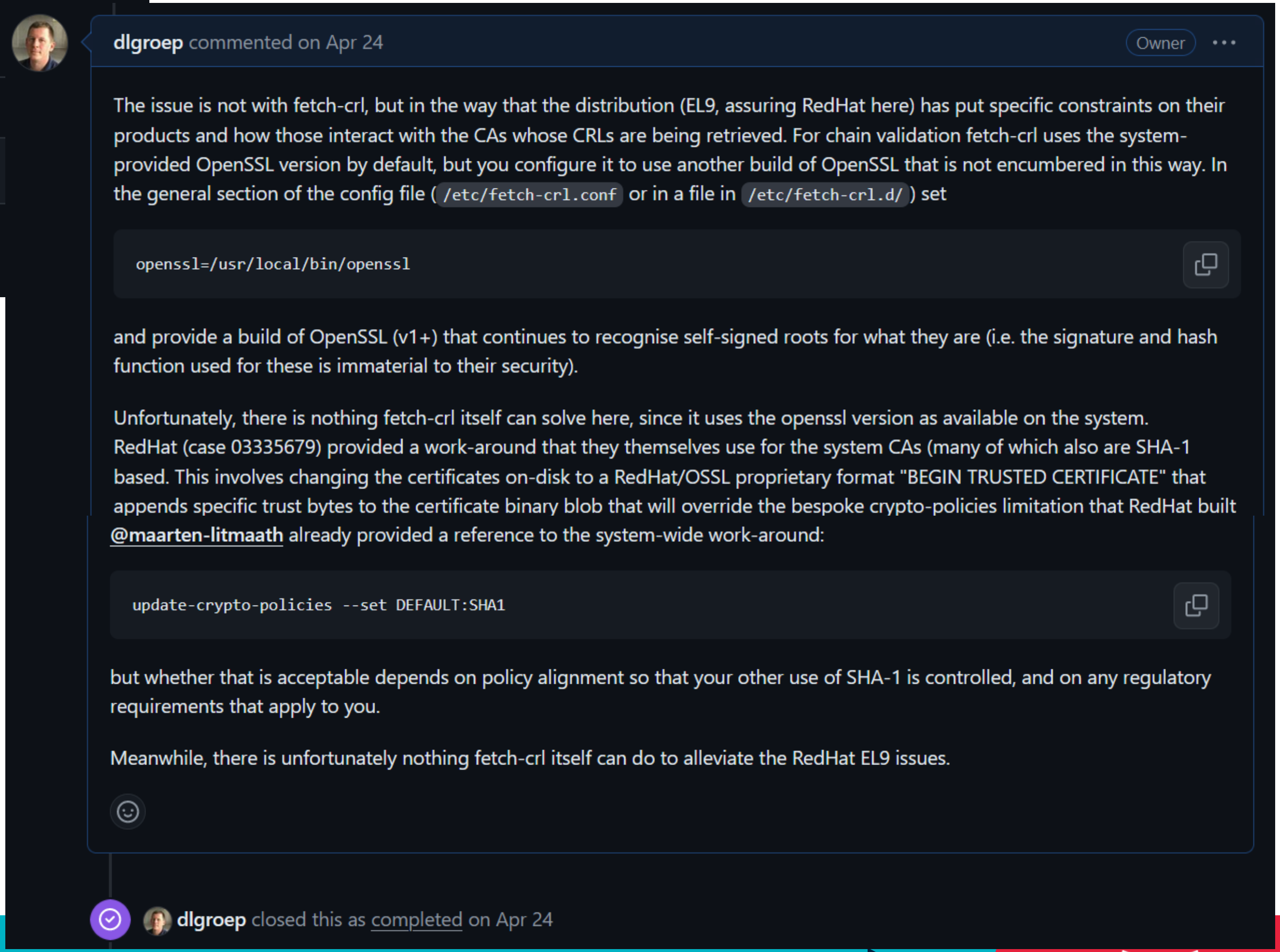


Fetch-crl doesn't work on EL9 with IGTF CAs #4

Closed vokac opened this issue on Apr 8 · 3 comments

[redacted] commented on Apr 8 • edited

I have clean installation of EL9



dlgroep commented on Apr 24 Owner

The issue is not with fetch-crl, but in the way that the distribution (EL9, assuring RedHat here) has put specific constraints on their products and how those interact with the CAs whose CRLs are being retrieved. For chain validation fetch-crl uses the system-provided OpenSSL version by default, but you configure it to use another build of OpenSSL that is not encumbered in this way. In the general section of the config file (`/etc/fetch-crl.conf` or in a file in `/etc/fetch-crl.d/`) set

```
openssl=/usr/local/bin/openssl
```

and provide a build of OpenSSL (v1+) that continues to recognise self-signed roots for what they are (i.e. the signature and hash function used for these is immaterial to their security).

Unfortunately, there is nothing fetch-crl itself can solve here, since it uses the openssl version as available on the system. RedHat (case 03335679) provided a work-around that they themselves use for the system CAs (many of which also are SHA-1 based. This involves changing the certificates on-disk to a RedHat/OSSL proprietary format "BEGIN TRUSTED CERTIFICATE" that appends specific trust bytes to the certificate binary blob that will override the bespoke crypto-policies limitation that RedHat built

@maarten-litmaath already provided a reference to the system-wide work-around:

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

but whether that is acceptable depends on policy alignment so that your other use of SHA-1 is controlled, and on any regulatory requirements that apply to you.

Meanwhile, there is unfortunately nothing fetch-crl itself can do to alleviate the RedHat EL9 issues.

dlgroep closed this as completed on Apr 24

https://github.com/dlgroep/fetch-crl/issues/4

But … maybe …

On 2023-12-20 13:25, Guido Pineda wrote:

> I am using fetch-crl version 3.0.22.
> We have a total of 89 trust anchors configured on our /etc/grid-security directory.
> I have tested fetch-crl with different versions of OpenSSL and here are the
> outcomes:
> For versions 1.1.1k and versions 3.2.0, the amount of errors when trying to verify
> the CRL's is only one [which was explainable]
> However, when using OpenSSL version 3.0.7, we get 10 errors

Due to self-compiling OpenSSL and does that ignore the RH crypt-policies?

Mitigations?

Still,

- 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 – state and progress

ASGCCA-2007
BYGCA
DZeScience
DigiCertGridRootCA-Root
KEK
MARGI
RDIG
SRCE
TRGrid

ArmeSFo
CESNET-CA-Root
DigiCertAssuredIDRootCA-Root
IHEP-2013
RomanianGRID
SiGNET-CA
seegrid-ca-2013

Fixed by now: GridCanada, CILogon basic/silver/OpenID, UKeScienceRoot-2007
Removed: DigiCertGridCA-1-Classic, DigiCertGridTrustCA-Classic, DFN-GridGermany, CNIC,
Pending withdrawal: LIPCA

Questions?

BUILDING OUR GLOBAL TRUST FABRIC

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

Nikhef
Maastricht University

this work is co-supported by the Trust and Identity work package of the GEANT project (GN5-1)

in collaboration with many, many people in the AARC+ Community, including Christos Kanellopoulos, Nicolas Liampotis, Licia Florio, Hannah Short, Maarten Kremers, Niels van Dijk, David Crooks, Dave Kelsey, Ian Neilson, Mischa Sallé, Jens Jensen, and so many others!

Thank you

davidg@nikhef.nl

GÉANT
Networks · Services · People
www.geant.org

This work has been co-supported by projects that have received funding from the European Union’s Horizon research and innovation programmes under Grant Agreemen No. 101100680 (GN5-1), 856726 (GN4-3), and 730941 (AARC2).