PROCEEDINGS OF SCIENCE

Tweaking the Certificate Lifecycle for the UK eScience CA

John KEWLEY*
Science and Technology Facilities Council (STFC) Daresbury Laboratory
E-mail: john.kewley@stfc.ac.uk

David MEREDITH
STFC Daresbury Laboratory
E-mail: david.meredith@stfc.ac.uk

Jens JENSEN
STFC Rutherford Appleton Laboratory
E-mail: jens.jensen@stfc.ac.uk

Obtaining and maintaining an X.509 certificate can be a significant barrier to non-technical users. Simple operations like renewing after expiry or changing an email address in a certificate are complicated, requiring additional meetings with RA Operators. The UK eScience CA is therefore modernising its software and policies to lower this barrier and reduce the support load on its helpdesk. To do this, we are developing a REST interface to the UK eScience CA with smarter client tools that simplify user and RA access. Improvements over the existing system include; a) renewals of recently expired certificates; b) a lightweight certificate re-application process that requires only a virtual meeting with an RA operator; c) a certificate change interface for amending selected attributes; and d) a service for bundling multiple host certificate requests into a single request. These changes will reduce the need for repeated face to face meetings with the user's RA and reduce tickets to the UK eScience CA helpdesk.

Keywords: Certification Authority, certificate, Grid, CA, RA, REST

EGI Community Forum 2012 / EMI Second Technical Conference,
26-30 March, 2012
Munich, Germany

*Speaker.

PoS(EGICF12-EMITC2)151

1. The UK eScience CA

The UK eScience Certification Authority (CA) is one of the world's busiest Grid Certification Authorities, having issued approximately 30,000 certificates. It is assisted by 93 Registration Authority (RA) Operator[s] at 66 institutions throughout the United Kingdom. It is accredited by the International Grid Trust Federation (IGTF) [1] and issues 13 month Grid certificates confirming to the Classic X.509 Authentication Profile [2].

Following IGTF policy, when certificates are initially requested from the UK eScience CA, the user must meet face to face with their RA Operator so their PhotoID can be checked and photocopied. Virtual meetings over Access Grid or Video Conference are not permitted. This is because the RA Operator must check the likeness on the PhotoID in person and also make a photocopy. Annual renewals[1] require only that the RA Operator confirms the user's continuing entitlement to a certificate: a face to face meeting is not required at this stage.

Until recently, the UK eScience CA has used a browser interface (based on an early version of OpenCA [3]) to request, renew and revoke certificates. However, for a variety of operational reasons we have had to make a number of bespoke changes to the software. As a result, we have been unable to keep up to date with the latest OpenCA release as reintegration would have become too time-consuming. Consequently, the version of our OpenCA has become out-dated and has become incompatible with the recent profusion of browser releases (even if more recent versions of OpenCA provide better support). As a result, there are now many browsers that we cannot support.

In fact, browsers generally make it harder for us to support certificates as they are designed for general certificate use and because they need to discourage users from trusting CAs that are not already in the browser keystore. For example, the private key is not usually available independently from its certificate and the certificate/key pair is exported in PKCS#12 [4] format which means users (often) need to extract the certificate and private key using arcane OpenSSL commands. In fact some browsers do not include the private key by default when exporting a certificate. Finally, browsers also tend to change how they support certificates over time.

Although this is highlighted on all our websites, we still receive a large number of UK National Grid Service (NGS) [5] helpdesk tickets where users have either overlooked or ignored these warnings. Furthermore, it is often non-trivial for the support team to identify when users have used an incompatible browser. This is because the certificate application process will fail at different stages, and with different errors depending on which unsupported browser has been used.

2. CertWizard and RESTful CAServer

In order to reduce such user issues and corresponding helpdesk tickets, we have developed CertWizard [6], a Java tool which is both operating system and browser independent. It can be deployed as either a WebStart application or an executable `.jar` file. It provides a consistent interface for users and assists in certificate application, renewal, revocation and installation of their certificate as separate `usercert.pem` and `userkey.pem` files. The modular design of the

[1] Technically when we say "renew" we really mean a rekey since the user creates a new key pair and a new certificate request and associates this new request with the old private key.

tool allows us to integrate it with MyProxyUploader, which provides an interface for generating Grid proxy certificates of various types, including those with attributes for Virtual Organization Membership Service (VOMS).

The CAServer provides support for both user and server certificates and has an additional interface for submitting bulk-host requests.

The CertWizard communicates with our newly developed CAServer over a secure Representational State Transfer (REST) interface. The REST service endpoints are secured and encrypted using a) HTTPS server authentication in conjunction with b) a proprietary Proof of Possession of Private Key (PPPK) challenge/response protocol which is used to authenticate the user. It is this server that then communicates with the database directly in a manner which is fully compatible with OpenCA which uses the database simultaneously (see Figure 1).

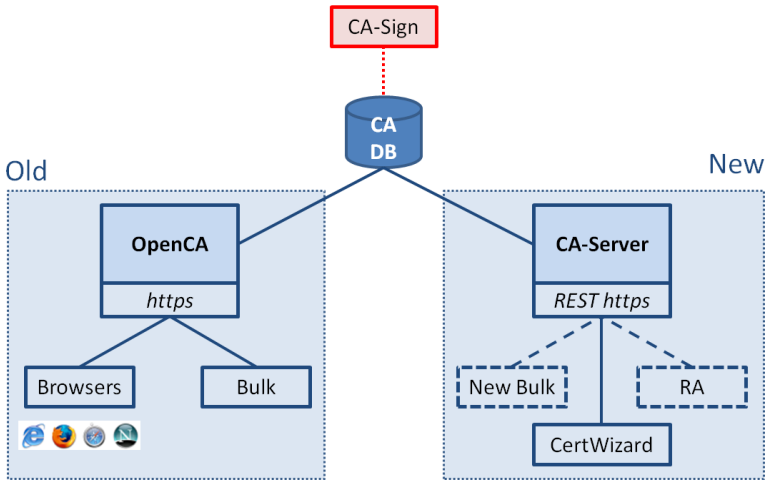

Figure 1: CertWizard and CAServer replace the previous Browser and OpenCA interface

This bulk-host request interface allows a large number of server certificates (belonging to the same domain) to be requested in a single request. The bulk-host client is currently a REST-enabled Perl script and is typically used by administrators of large Grid resources. In the future, we also envisage that the bulk request service may be utilised by our training CA for requesting a set of training certificates.

3. Current UK eScience Certificate Lifecycle

Figure 2 shows the current UK eScience Certificate Lifecycle as a State Transition Diagram (STD). Its is typical of many CAs and hence some of our suggested changes later are widely applicable. The ovals show the three certificate states while the boxes show the UK eScience CA processes that are used to create a valid certificate (Apply and Renew) or change its state (Revoke). Certificates expire naturally without any user-invoked process so that transition is left unlabelled. Once approved and signed, a certificate remains valid until it is renewed, revoked or automatically expired after 13 months. In the latter two cases it can no longer be renewed and the user must apply

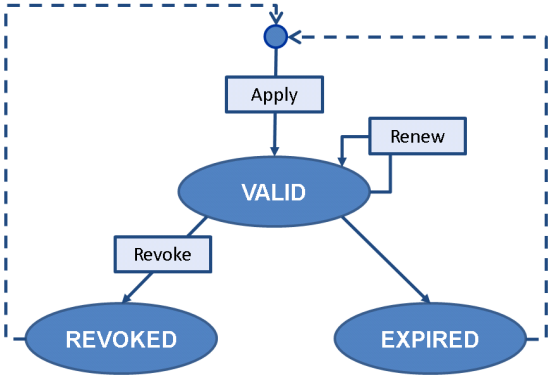

Figure 2: State Transition Diagram of the Current UK eScience Certificate Lifecycle

for a new certificate (this is represented in Figure 2 by the dotted lines): this would entail a further face to face meeting with the user's RA Operator. Subsequent figures will show amendments to this original lifecycle in red.

3.1 Problems with the Certificate Lifecycle

The NGS Support Centre receives many tickets from users who can no longer use the Grid as it turns out that their certificate has expired without them realising: one day it works; the next day it doesn't. There can be several reasons for this: maybe their site spam filter prevented our reminders from reaching them; maybe their email address has changed; or maybe it has gone to a mailbox that they only check infrequently. If a user's email changes during their certificate's lifetime, then their only available option is to request the revocation of their current certificate and apply for a new one with the new interface which provides a method for amending a certificate's email address at renewal.

Our policy (in accordance with IGTF policy) of insisting on face to face meetings for new certificate requests works well unless users miss their renewal date, forget their password, need to change their certificate's email address, or have successfully applied for a certificate only to find they used the wrong browser and now cannot download it. In theory, a visit to a local RA Operator shouldn't present much of a problem, but not all institutions have their own RA and many of our users spend time on secondment at other institutions (CERN for instance). Visiting an RA Operator therefore, can be both time-consuming and expensive, especially as it is only to re-check a PhotoID that has already been checked and photocopied on a previous occasion.

4. Changes to the Certificate Lifecycle

To counter the problems mentioned above we are planning the following changes to our Certificate lifecycle by amending the associated software and policies. Note that these have been described as separate changes, but in Figure 6 they are combined.

4.1 Re-Applications

Since a valid certificate is needed for the renewal process, if a user's certificate has been revoked, lost or expired then renewal is not possible. Instead, the user must follow the process for requesting a new certificate. Currently this is also the case if a user's email address has changed or a certificate has been lost, although in these cases the current certificate must first be revoked. What the user wants to do is re-apply for a certificate without attending another face to face meeting with their RA Operator. It is this process of Re-applying that we want to simplify: reducing involvement of the RA Operator and eliminating the need for any subsequent face to face meetings after the initial one.

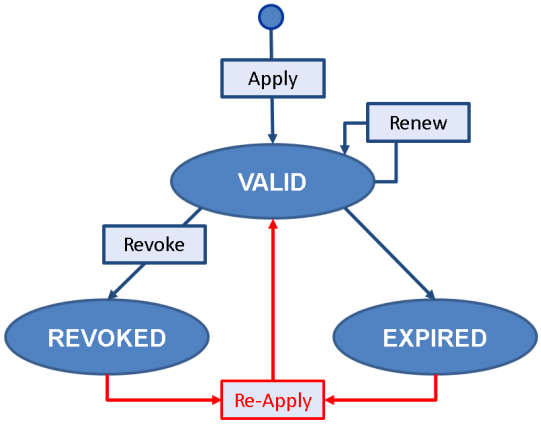

Figure 3: Introducing the Re-Application Process

What we require is a "Repeat Application" process that lies somewhere between an initial application and a renewal. We call this a "Re-Application". Figure 3 show how the proposed re-application amends the existing User Certificate Lifecycle. As illustrated, we propose that in a re-application the user only needs to meet virtually with the RA Operator rather than face to face. This can be carried out over Access Grid or Video Conference. The rest of the process is largely the same as an initial application, but the key difference is that the RA Operator needs only confirm a user's on-screen likeness using the photocopy of the user's PhotoID (taken during the initial face to face meeting). The RA Operator should also confirm that the user has the same identifying numbers on his/her PhotoID. The remainder of the Approval process is identical to a New request. Therefore, for auditing purposes, the RA Operator must still make a record of that virtual meeting.

4.2 Permit Renewal (Rekeying) of Recently Expired User Certificates

Following the lead of the IGTF-accredited DOEGrids CA [7], we would like to allow users to renew a recently expired user certificate. This amends our certificate lifecycle as shown in Figure 4. This will require changes to the CertWizard software and also amendments to our CA policy. The "grace period" cannot be too large since users and RA Operators tend to be more casual about revoking lost or stolen certificates if they have already expired. We have therefore decided to follow the precedent set by the DOEGrids CA and allow a 30 day grace period.

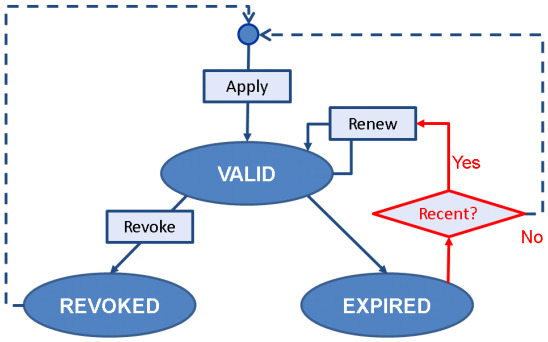

Figure 4: Renewing a Recently-Expired Certificate

Note that we have been running an analogous renew-after-expiry scheme "manually" for some years now: people who need to "renew" their certificate after expiry request a new certificate with the same distinguished name as the old one, and then use the old private key to sign an email containing the PIN used to request the new certificate. A CA operator then checks that a) the names match, b) the PIN matches the request, and c) the signature is valid in every respect except for the fact that the signing certificate has expired. Once the operator is satisfied, they may then approve the new request and archive the email. The archived email can be used to prove to an auditor that the person with the new certificate was the same as the person with the old certificate.

Few mail user agents allow email to be signed with expired certificates so we have implemented a basic recipe which involves creating the signature with OpenSSL and piping the mail into `sendmail`. This works only on UNIX® systems and similar; it is therefore an option we only use for the more technically minded users and/or the most urgent cases.

The new protocol associated with renewing expired certificates works essentially the same way: it proves possession of the private key associated with the now expired certificate. This could have been implemented in two ways: a) by calling normal signature validation routines, but modify them to overrule (in this case) the fact that the certificate has expired; or b) implement the PPPK protocol independently, using the normal RSA[2] challenge/response.

The original design for PPPK was not just for certificates but also for requests (e.g. to delete requests, or prove rights to access the certificate). We therefore chose the second option (b) using our own key check of RSA keys with an algorithm very similar to the normal key challenge/response except based on the public key only, rather than the certificate. Of course some actions still require a valid certificate to be present: RA operations for example.

4.3 A Change Interface

We are also considering a separate Change interface for amendments to certificates that do not require a full renewal or re-application, such as a change to an Email address for example. This is illustrated in Figure 5. As neither the DN nor the lifetime of the certificate would change, it is feasible that these change requests would not require the approval of an RA Operator. A Change

[2]A public key algorithm by Rivest, Shamir and Adelman [8]

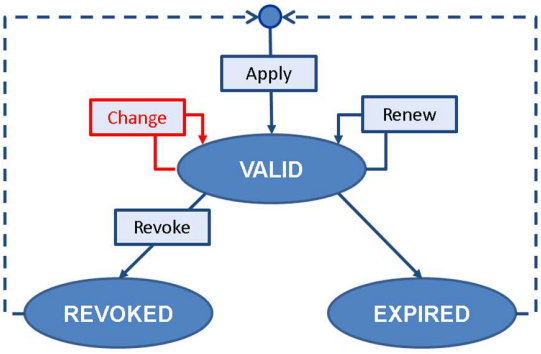

Figure 5: Certificate Lifecycle with Separate Change Process

interface could also be extended to amend other certificate details, such as adding Code Signing attributes. However, if the CA policy states that RA Operator approval must extend to cover such changes (including amendments to user email address), it would be more sensible to combine the features of the change interface with the renewal process. In doing this, a user would be able to amend certain certificate attributes only when renewing. The final implementation of the change interface is currently being discussed within the UK CA team.

4.4 Putting it all together

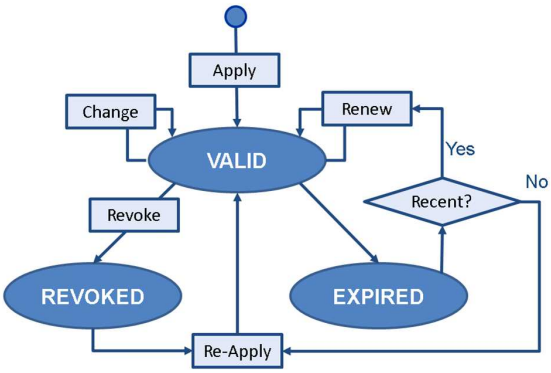

Figure 6: Proposed UK eScience User Certificate Lifecycle

Combining all the above lifecycle changes together gives us Figure 6. These improvements are a) renewal of recently expired certificates; b) a lightweight certificate re-application process that can be satisfied with only a virtual meeting with an RA; and c) a certificate change interface for amending selected attributes (and in particular the email address associated with the certificate).

5. Future Work

We are currently completing support for both Single-Host and Bulk-Host certificate requests in the CAServer and in our REST clients. Once completed, our next major development will centre on

the design of new REST endpoints for performing RA Operator tasks and corresponding client tool development. Like the OpenCA portal, the new RA operator interface will support parameterised searches of the CA database for different types of request, and subsequent approval and/or deletion of those requests.

Importantly, by providing a well defined set of REST services, different client tools can evolve accordingly. For example, we aim to extend our Java CertWizard with an RA operator interface. Conversely, we do not intend to extend our Perl script with the RA Operator functionalities — it will only support bulk-host certificate requests, unless demand dictates otherwise. It is also feasible that we design a new Web portal interface by re-using the APIs developed for use in the REST CAServer, that is, by reusing the abstractions that lie directly beneath the REST endpoints, such as the transactional service façades and data access objects. It is worth noting that a portal would not use the PPPK secured REST endpoints, but instead would use regular client-authentication using browser certificates.

We are also aware of other groups who have developed similar interfaces to their CAs and aim to engage with those groups to investigate the scope for collaboration. This could include code and software security reviews, and potentially a trial deployment within other CAs. In doing this, the software's usefulness and scope for reuse within projects such as the EGI can be better judged. Such feedback would be invaluable in highlighting missing functionality and identifying additional abstractions that would no doubt be required for subsequent CA customisation. While it is unlikely that that the software would fully replace an existing CA system, a feasible deployment scenario for other CAs would be to host a version of the REST services in conjunction with their current system as shown in Figure 1. Those CAs could then provide a modified version of the CertWizard for their CA and/or provide additional clients to their CA. At the very least, we will aim to identify common modules that can be abstracted and packaged for reuse (e.g. the PPPK protocol and our implementation which is largely standalone and requires only a single database table).

Directions for further work also include an advanced interface to request robot or code-signing certificates, or for when users have their own hardware tokens.

6. Conclusions

The UK eScience CA helpdesk receives many tickets relating to certificates and their renewals. This is largely due to browser incompatibilities with our deprecated OpenCA Web interface. We are therefore improving our certificate lifecycle with the introduction of new and streamlined processes that will make it easier for users to manage personal and host certificates. To do this, we are developing a REST interface to the UK eScience CA with smarter client tools that simplify user and RA access.

By making the improvements to our X.509 certificate lifecycle shown in Figure 6, combined with the ability to request multiple host certificates as a bulk request as shown in Figure 1, we aim to reduce the need for reapplications and also provide an alternative to the currently required face to face meeting for renewals. Users will therefore be able to renew recently expired certificates, amend their certificates email address and avoid unnecessary face to face meetings when on secondment. Although these improvements require extra development effort, we can subsequently take control of our certificate lifecycle and improve our users' experience.

7. Acknowledgements

We'd like to Acknowledge the support of our STFC and NGS colleagues as well as the NGS funding from JISC.

References

[1] International Grid Trust Federation: `http://www.igtf.net`

[2] Authentication Profile for Classic X.509 Public Key Certification Authorities with secured infrastructure `http://www.eugridpma.org/guidelines/classic`

[3] The Open CA Project: `http://www.openca.org`

[4] *PKCS 12 v1.0: Personal Information Exchange Syntax* `ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-12/pkcs-12v1.pdf`

[5] NGS: `http://www.ngs.ac.uk`

[6] J. Kewley, D.J. Meredith, J. Jensen, *CertWizard: a New Certificate Tool for the UK NGI User Community*, presented at *EGI Technical Forum 2011*, Lyon, France, 19-23 Sep 2011, `http://epubs.cclrc.ac.uk/work-details?w=61855`

[7] M. Helm *et al*, *DOEGrids CA Certificate Policy and Practice Statement*, Version 2.1, August 2009, `http://www.doegrids.org/Docs/CP-CPS.pdf`

[8] RSA Laboratories *PKCS #1 v2.1: RSA Cryptography Standard*, June 2002, `ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-1/pkcs-1v2-1.pdf`