Are personal digital certificates really usable and scalable?

Mark Norman[*], Alun Edwards, Christian Fernau

Research Technologies Service, Oxford University Computing Services

Abstract

This poster outlines the findings of the Digital Certificate Operation in a Complex Environment (DCOCE) project that has recently concluded at Oxford University. PKI is used as a basis for security of the UK e-Science Grid. It was therefore important to ascertain whether the use of digital certificates in higher and further education is scalable to more than a select number of technical users. The project explored the advantages and disadvantages of end user/client digital certificates as means of on-line authentication in a higher or further education information environment. We conclude that the use of client certificates is feasible and scalable. With usability feedback from over eighty users, with a broad spectrum of technical abilities, the DCOCE project looked further into feasibility issues than most other studies where a common desktop environment does not exist. The DCOCE project developed and tested an alternative model of PKI whereby user data was held largely within the institution at a central Registration Authority (central RA). Certificate requests are held by the central RA and passed on to an external Certification Authority (CA). Thus, user data are kept close to the users and the CA specialises in the high availability service of generating/signing certificates and managing revocation lists. This model should prove far more scalable. Certificates could also be useful to some users as the front-end authentication tokens for single sign on systems and we believe that it is not critical that most users will never fully understand how they work. Making the system of issuing and renewing the certificates as user-friendly as possible appears to be the most critical factor.

1. Introduction

The Digital Certificate Operation in a Complex Environment (DCOCE) project was a two-year Joint Information Systems Committee (JISC) funded project that completed in early 2005.[1] The main aim of the project was to look into the use of digital certificates by end users in higher and further education (HE/FE) for authentication to services and also to look at the methods of issuing certificates to users and how to manage the 'accounts'.

This short paper accompanies a poster of the same title presented at the UK e-Science All Hands Meeting 2005. Much of this text has been presented elsewhere and is a summary of a larger body of work.[2] [3] [4]

Digital certificates are the main end-user security credentials for UK e-Science. Many consider digital certificates to be problematic for the 'average' user in terms of usability and, consequently, scalability.[5] Conceptually, the use of digital certificates could enable authentication to more than just grid applications. These could include:

- restricting access to a department web site, or set of pages;
- restricting access to an on-line service, such as a database or library catalogue;
- authenticating reliably to a distributed service - such as Eduserve's Athens - as an alternative to username and password;
- providing an alternative to the common, but problematic IP range restriction, used by many institutions and libraries.

Therefore, the potential of using client digital certificates more widely can clearly be seen.

The DCOCE project developed most components from scratch and also built up a near 'classical' PKI policy regarding the use of multiple Registration Authorities (RAs) within Oxford University. However, we deviated from a classical PKI design as we made all of the RAs subordinate to a central RA. Certificates were issued by GlobalSign (the Certification Authority - or CA) but the architecture of the PKI was such that any commercial or non-commercial (or internal) CA could be used.

[*] Author for correspondence

2. How the PKI worked

2.1. Policies and practices

The issuing of certificates to staff and students was performed to set procedures (or certification practices), as with most registration or account creation tasks. One strength of PKI is that it is accompanied by a legal or pseudo-legal certification practices statement that puts constraints upon the procedures that may be used or even details the exact procedures that may be used.

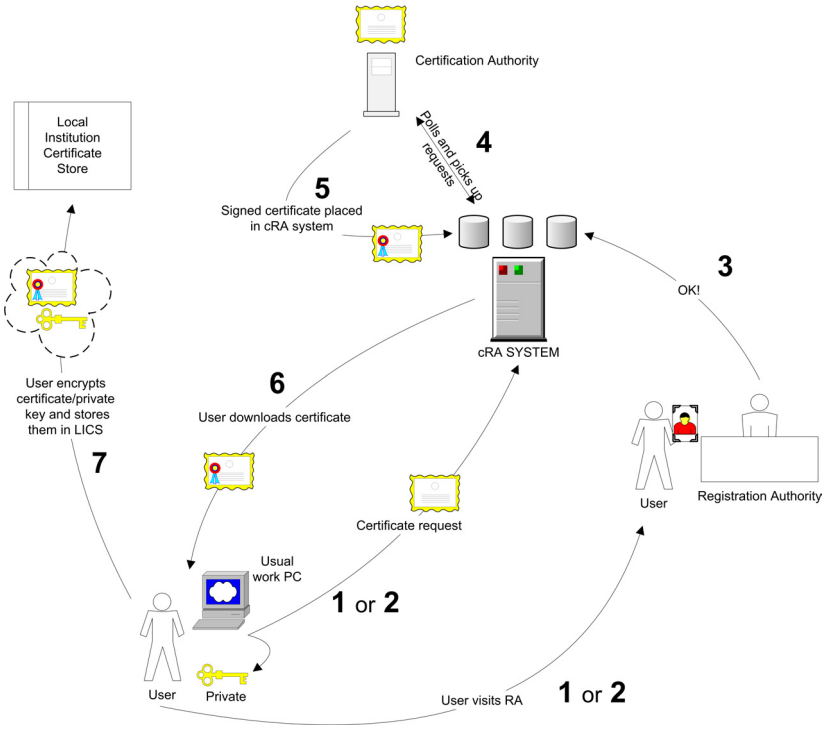

Figure 1. Procedures for requesting and downloading a certificate.

2.2. The certificate request/download cycle

In the DCOCE PKI, a user would typically make a browser-based request, before visiting the RA for her organisational unit (OU). The OU is typically a college, department or defined group and the applicant's RA will have means of checking her membership of the OU. Her RA also uses a browser to see that the user has made a request and checks that the applicant is a current member of the OU. After the RA has seen some form of photo ID (usually a university card), he then approves her certificate request. This procedure is outlined in Figure 1 as stage 1 or 2 (with the alternative procedure outlined later). There is a central database where the requests and a little personal information are held. The requests are batched up and submitted to the external CA, who generates certificates and returns them to the central database. The applicant either receives an email (if she provided an email address) or merely visits the central database via a web interface after a short time. This allows her to download her certificate. Her certificate is reunited with her private key, which was generated at the time of the request (and must remain available or secret to the applicant only).

Once the certificate has been downloaded and installed in her web browser, it is available to be used and to gain access to on-line services.

As mentioned above, there is an alternative certificate-issuing procedure that serves users who visit their RAs before making a request from their computer. The users may have to visit their RAs to pick up their university cards and it is unreasonable to insist on two visits. Therefore, the RA checks the applicant's details and issues her with a nine-digit code so that she can make a request from her own computer at a later time. That request is said to have been 'pre-authenticated' and is approved automatically.

(Step 7, as illustrated in Figure 1, is described in section on *The Local Institution Certificate Store*, below.)

3. Main findings

3.1. Client usability

In general, the use of the client digital certificates was quite straightforward for most users. However, there were many minor problems regarding usability of the certificate *issuing* mechanism. Nevertheless, there were no 'showstoppers' that could have meant that the widespread use of client certificates was unfeasible in a HE/FE context.

We did not try to educate users in the difficult mathematical basis of public/private keys: something that technically-minded people

often need to spend a while working out, before they trust the concept. We found that the ideas of keeping a pass phrase secret and of having a 'file' on their computer (or hardware token) were accepted. However, we believe that these two ideas will have to be brought out more strongly than our experimental interface suggested for the majority of users to be aware of them.

Once the certificates were installed, the usability reports were overwhelmingly positive.

3.2. The use of cryptographic hardware tokens

Despite some Linux and MacOS implementation issues, we believe that the use of cryptographic tokens is highly attractive to a scalable PKI for HE/FE institutions. We evaluated a USB device supplied by SafeNet and the findings are documented in our report.[1] There is far less for the end user to do and understand. As long as she has the token in the port and she has supplied (a relatively insecure, and easy to remember!) pass phrase, it all works. At the current levels of technology, the tokens are adequately secure from the attentions of hackers. As well as the easy, intuitive nature of the use of the tokens, the issuing mechanism is far simpler as well. Technical end users - or those requiring high-level assurance - could still carry out a browser-based request but the vast majority of users could be issued tokens that are pre-loaded. This could be carried out on a large scale and is still secure from a rogue sysadmin as the private key never leaves the token: nor can it be read or exported.

3.3. The Local Institution Certificate Store

One of the usability difficulties for the non-technical end user is the difficulty of moving his certificate if he moves to another computer, or if his usual computer is re-built etc. For this reason, a centralised backup is desirable. However, it goes against some of the principles of PKI that a private key should exist elsewhere, or for it not to be under the exclusive control of its owner. Balancing these considerations, we designed an architecture that we called the Local Institution Certificate Store (LICS). For those users who opted to keep a backup, the LICS contained their certificate and an encrypted version of their private key. The private keys are encrypted by the users, using the pass phrase of their software key store (the 'software security device' in Mozilla/Netscape).

With regard to security, an attacker who does not know a user's pass phrase would first have to guess the hash (to receive the encrypted private key and certificate) and then to run an off-line attack on the encrypted private key. Since knowing the hash should provide no benefit in guessing the pass phrase and therefore breaking the encryption, this system was deemed to be secure, even against an attack by a rogue sysadmin who would already have access to the 'password' file containing the list of hashes.

Unfortunately, since establishing this design, we have learned of the time-memory trade-off attack, whereby an attacker with access to the list of hashes is able to run an off-line attack and obtain the pass phrase, especially for those private keys encrypted with weaker or more common pass phrases. Therefore, it is likely that our LICS is, after all, vulnerable to attack by a determined rogue sysadmin. However, as many authentication systems always implicitly trust their own sysadmins, this is not a great disadvantage and we are still hopeful that a modification of our system entailing salting could solve our problem. Clearly, a determined hacker who obtained the hash 'password' file would also pose a threat, but this can be defended against using the usual defences of good system administration and later detection could result in the revocation of all of the certificates.

In conclusion, our LICS - made use of by the majority of users - is not as secure as we had, at first, hoped. However, it is secure enough to protect the kind of resources that these certificates are to be used against, and more secure than most 'systems' used to protect those kinds of resources at present.

At present, we would prohibit the use of the LICS and no central backup would be taken, were we to issue certificates to protect financially valuable resources (such as accounts databases etc.). Our project aimed to issue basic level assurance certificates, which would be inappropriate for highly valuable or sensitive resources and the use of the LICS is quite satisfactory for such a purpose.

3.4. Authentication and authorisation

Conclusions regarding authentication were made and these are too many to list here. However, one interesting 'quick win' regarding digital certificates was discovered regarding *implicit authorisation*. This would seem to be of relevance to this audience. We found that certificates were very attractive to OUs within the University (the *Organisation*) in that, with very little effort, they could be used to allow only members of that OU to access

departmental or college web pages. Figure 2 shows a snippet of a configuration file that allows an Apache server running mod_ssl to only allow access to members of Oxford University Computing Services ("oucs"). This is incredibly simple and means that the web site owner does not have to worry about maintaining lists of authorised users. This is a major saving in effort.

```
SSLRequire %{SSL_CLIENT_I_DN_CN}
    == "GlobalSign PersonalSign
        Class 1 CA" \
    and    %{SSL_CLIENT_S_DN_O}
    == "Oxford University" \
    and    %{SSL_CLIENT_S_DN_OU}
    == "oucs"
```

Figure 2 Apache server/mod_ssl code filtering for organisational unit

4. Conclusions

The PKI-related conclusions of our project include the following, that:

- the use of PKI and client certificates is feasible and scalable;
- users do not need to understand the esoteric nature of public/private keys - they merely need to understand that there is something that needs to be kept secret, but available on their computer, for the procedure to work;
- cost-effectiveness could become an issue in institutions that support multiple operating systems and software, if relatively few certificates are to be issued (but cryptographic hardware tokens could be used to mitigate for this);
- the use of cryptographic hardware tokens to hold each user's private key and certificate are highly desirable as they ease usability and scalability by a great degree;
- hardware tokens that are cryptographically secure enough are probably too expensive at present and there are currently some operating systems compatibility issues to be resolved;
- the use of client digital certificates can make some common authorisation problems trivial to overcome;
- authentication and authorisation should be separated as much as possible (despite digital certificates being able to accommodate authorisation information within their fields);
- it makes better sense to store authentication digital certificates (and private keys) in the operating system, rather than in software.

Other conclusions from the project also include:

- the system of using RAs for account creation triggering/activation and the issuing of authentication tokens is highly desirable where the RAs are based within the same OU (department or subunit etc.) as the applicant;
- central registration staff, and especially sysadmins should not play a direct role in authenticating individuals for accounts and issuing authentication credentials;
- such central staff should, however, police the RAs and the database in order to counteract fraud and mistakes.

And will client digital certificates ever take off in the information environment (i.e. more widely than with grids)?

- the future use of client digital certificates in the HE/FE sector is closely related to the fortunes of single sign-on (SSO) initiatives and the Shibboleth development;
- we believe that integrating client digital certificates with SSO provides a solution to bridge any gaps that appear between SSO and Shibboleth, especially for those users accessing services outside of their own institution and/or Shibboleth federation.

5. References

[1] The DCOCE project. http://www.dcoce.ox.ac.uk

[2] Norman, M.D.P. 2004. Accessing services with client digital certificates: a short report from the DCOCE project. *New Review of Information Networking,* Vol. 10(2): 193-15.

[3] Norman, M.D.P., Fernau, C., Edwards, A. 2005. Will client digital certificates ever fly? A short report from the DCOCE project. *Proceedings of the UK Unix and Open Systems User Group*.

[4] Norman, M.D.P., Edwards, A, Fernau, C., Sytsema, J. Wilson, J.A.J. 2005. The DCOCE Final Implementation and Evaluation report http://www.dcoce.ox.ac.uk/docs/

[5] Beckles, B. 2005. Overview of local security issues in Campus Grid environments. *Procedings of NeSC Campus Grid meeting*, June 2005. http://www.nesc.ac.uk/talks/556/5_Beckles_Security.ppt