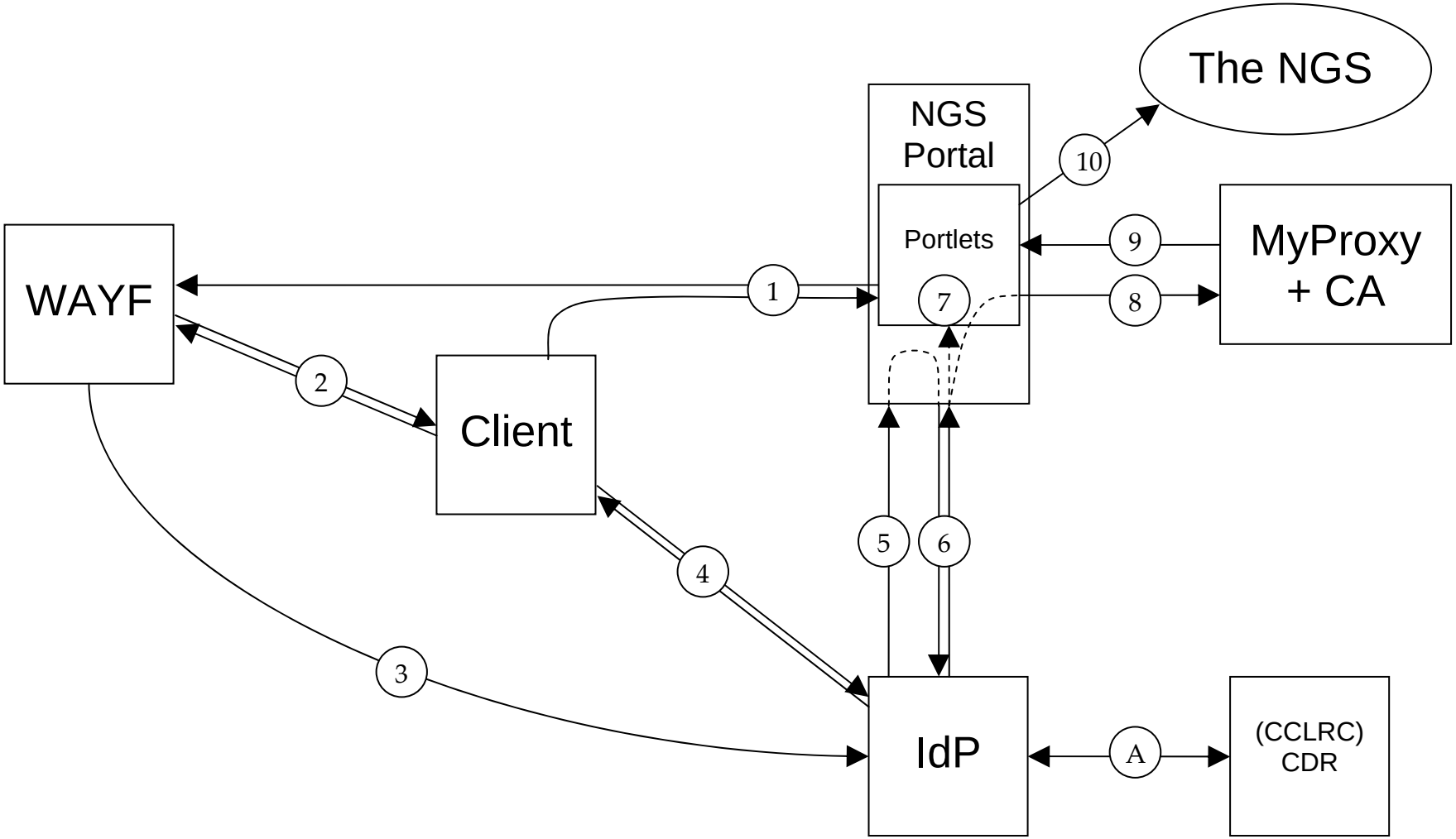
The NGS
NGS Portal
Portlets
MyProxy + CA
WAYF
Client
IdP
(CCLRC) CDR
1
2
3
4
5
6
7
8
9
10
A

Notes of Shibboleth Implementation

1. User requests access to the NGS portal (the SP) through a Shibboleth logon, the user's browser is redirected to the WAYF.

2. The user chooses the appropriate IdP in the form returned by the WAYF.

3. The user's browser is re-directed to the correct IdP.

4. The user is authenticated by the IdP, via user of CDR (A).

5. The IdP redirects the user's browser back to the NGS portal (SP). The signed authentication SAML assertions are passed in this redirect, via user of CDR (A).

6. The NGS portal calls out to the IdP's Attibute Authority for attiributes about the user. In this case we envisage username (for retrieving previously uploaded proxies), real name (for DN creation) and project.

7. The user is authorised to use the NGS portal through these attributes.

8. In addition the signed attribute assertions are additionally signed by the portal and passed on to the MyProxy server. These credentials are used as either the authorisation to release a proxy (if the user has already uploaded a proxy) or as authentication to allow the auto-generation of a low assurance proxy, from the built in MyProxy-CA.

9. The proxy credential is returned, optionally other attributes can be added to the proxy (through the use of certificate extensions). Most notably a vo attribute can be added based on a project attribute returned in the attribute assertions. This can be used by gatekeepers to restrict access to resources.

10. The user can access the NGS.

Where is the work?

1. The apache shibboleth module needs to be modified to pass on the signed attribute assertions (they are not signed at the moment).

2. The NGS portal needs to use the shibboleth module for authentication, gleaning username information from the module and needs to sign and pass on the attribute assertions.

3. The MyProxy server needs to be modified to accept signed attribute assertions for authorisation and username->DN mapping in the CA. (Note similar work to this has happened in the GridShib project – code re-use maybe be possible).

Notes on MyProxy setup

Low-assurance DNs will be of the form:

/C=UK/O=eScienceShibboleth/OU=<Organisation>/L=<Location>/UID=<username in organisation/location>/CN=first-name last-name

The CN part ensures that in the majority of cases the CN will be the same as a normal eScience certificate CN, which can be important for SRB access. The UID field (could be named anything) is used to ensure uniqueness.

The MyProxy setup is the same as the setup used in the CCLRC SSO project: two myproxy services will be running on the same machine with a common proxy storage directory. One will be a normal instance of myproxy, which users can upload proxies to (using the myproxy-init command with the –n/--no_passphrase option) and the second will be the shibboleth interface used by the portal to retrieve the credentials. The shibboleth version will also be configured to act as a low assurance CA.

Note that as users will be storing proxies with the –n option so will not be able to retrieve them through the standard interface as their will be no way of authenticating them (unless we provided a username/password database or a link to the CDR like we have in the CCLRC SSO project).