

LazySession

The Shibboleth ServiceProvider supports a feature called `LazySessions` that takes the place of a more typical programming-language-level API for invoking its protection. In most deployments, session setup is handled automatically by configuring rules based on resource URLs or with platform-specific web server configuration. This allows applications to be written without any dependencies on the particular behavior of Shibboleth, and simply consume data pushed into environment variables like `REMOTE_USER` and others. Many applications can run unmodified behind Shibboleth or any other SSO system that has a similar environment variable interface.

For those situations where a finer degree of control is required, however, an application can make use of the services of a `SessionInitiator` to ask for a session to be established from within the application code. Instead of implementing to a language-specific API, a simple HTTP-based protocol using redirection and a query string is used to invoke the `SessionInitiator`.

Because the session is established later in the application dialog with the user, or potentially not at all, the term "lazy" was coined to refer to the mechanism.

In all respects, a `LazySession` is identical to a normal required session, except that in the case of a timeout or session expiration, the application is expected to detect the absence of a session (based on the lack of information supplied to it in the environment) and take steps to re-establish the session by repeating the original process again, if it wishes to.

Also note that no `AccessControl` is possible without a required session, this is also left to the application.