Configure Website for Two-Factor Authentication in Apache

Two-factor authentication requires a user to log in with a username, password and a second factor, such as a Duo two-factor option. You can configure your website to require two-factor authentication to provide greater security for your service.

Apache 2.4

```
AuthType shibboleth
ShibRequestSetting authnContextClassRef https://refeds.org/profile/mfa
ShibRequestSetting requireSession 1
<RequireAll>
    Require authnContextClassRef "https://refeds.org/profile/mfa"
    Require shib-session
</RequireAll>
```

Apache 2.2

```
AuthType shibboleth
ShibRequestSetting authnContextClassRef https://refeds.org/profile/mfa
ShibRequestSetting requireSession 1
ShibRequireAll on
ShibCompatWith24 on
Require shib-session
Require authnContextClassRef "https://refeds.org/profile/mfa"
```

If this site only require Two-Factor for certain location, this configuration will not work reliably. If the user doesn't have valid session and requests content in the Two Factor required directory first, two-factor will be enforced. If the user requests content from your site that NOT requires Two Factor and then requests content in the `Two Factor` directory, user may get authorization denied error if user hasn't completed two factor.

To resolve this issue, we can redirect authorization denied error to a script and use that script to redirect user to IDP for second factor authentication.

<Location /myTwoFactor>

```
AuthType shibboleth
ShibRequestSetting authnContextClassRef https://refeds.org/profile/mfa
ShibRequestSetting requireSession 1
<RequireAll>
   Require authnContextClassRef "https://refeds.org/profile/mfa"
   Require shib-session
</RequireAll>
```

ErrorDocument 401 /cgi-bin/mfaChk.cgi (this is just an example, you can replace it with your own script)

</Location>

mfaChk.cgi

```
#! /usr/bin/perl
use strict;
use CGI qw(:standard);

#get user's authentication context. If user already have MFA profile in the context,
#it means user is redirected to this script because user doesn't meet other authorize rules. So we
should just redirect user to an error document.
#if mod_proxy_ajp is being used in your environment, you might need to replace $ENV
{Shib_AuthnContext_Class} with $ENV{AJP_Shib_AuthnContext_Class}
my($authnContext) =  $ENV{Shib_AuthnContext_Class};
if ( index($authnContext, "https://refeds.org/profile/mfa") > -1 ) {
    print redirect(-url=>'/error.html');
}else{
#user hasn't completed 2FA, send user back to IDP for second-factor authentication
    my($returnTarget) = $ENV{REDIRECT_SCRIPT_URI};
    print redirect(-url=>"/Shibboleth.sso/Login?authnContextClassRef=https%3A%2F%2Frefeds.org%2Fprofile%
2Fmfa&target=$returnTarget");
}
```