Enabling OCSP validation in Apache Server:

Public release of Apapche 2.4.43 or greater is recommended; First verify what version you are working with.
To avoid any confusion, looking in the ~Apache_Version/modules/ssl/mod_ssl.c "GOOD" version will have the following line:

```
SSL_CMD_SRV(OCSPEnable, RAW_ARGS,
        "Enable use of OCSP to verify certificate revocation mode ('on', 'leaf', 'off')")
```

Where the "BAD" version or older version will have the following line:

```
SSL_CMD_SRV(OCSPEnable, FLAG,
        "Enable use of OCSP to verify certificate revocation ('on', 'off')")
```

Use case 1: Here OCSP Responder Signer Certificate in this case is issued by the same Root as CAC/PIV certificate; multi-Tier PKI (Root – Intermediate – Client), this specific OCSP responder only includes Tier 3 template info, i.e., information for the client certificates.

For this case, where we can not process OCSP responses for any certificates in the chain other then the client itself, option is available to use **SSLOCSPEnable** directive. This option can be set to 'on', 'leaf', and 'off'. When set to 'leaf' OCSP responder will validate client certificate itself only, not the intermediate or root.

It is recommended that this option is combined with a responder selection option **SSLOCSPDefaultResponder** URI directive and **SSLOCSPOverrideResponder** on/off. These two options in "URI + on" form will allow to use a pre-set OCSP responder other then one specified in the certificate itself.

For user case 1, OCSP configuration will look like this:

```
SSLOCSPEnable leaf
SSLOCSPDefaultResponder "{OCSP Responder URL}"
SSLOCSPOverrideResponder on
```

Use case 2: Here OCSP Responder Signer Certificate in this case is issued by the same Root as CAC/PIV certificate; multi-Tier PKI (Root – Intermediate – Client), OCSP Responder includes certificates for the entire PKI infrastructure.

For user case 2, OCSP configuration will look like this:

```
SSLOCSPEnable on
SSLOCSPDefaultResponder "{OCSP Responder URL}"
SSLOCSPOverrideResponder on
```

Use case 3: Here OCSP Responder Signer Certificate is issued by the DIFFERENT (other then the Root) CA as CAC/PIV certificate; multi-Tier PKI (Root – Intermediate – Client), and it only includes configuration for the client certificates.

It has to be noted that certificates for the Intermediate and Root CA for our CAC/PIV certificate are not longer required in the **SSLCertificateChainFile** directive. It became obsolete with version 2.4.8 (2.4.08) . Currently **SSLCertificateFile** directive is used to load intermediate CA certificates from the server certificate file in addition to root certificates.

For our configuration, **SSLCertificateFile** (in .pem format) must include root and intermediate CA certificates for the client certificate being OCSP validated.
Further, since OCSP Responder signer certificate is not issued by the same Root as certificates being validated, we must specify the OCSP Certificate file location using Apache "**SSLOCSPResponderCertificateFile** file". This directive option supplies a list of trusted OCSP responder certificates to be used during OCSP responder certificate validation. The supplied certificates are implicitly trusted without any further validation. This is typically used where the OCSP responder certificate is self signed or omitted from the OCSP response.

For use case 3 certificate only OCSP validation, configuration options will look as follows:

```
SSLOCSPEnable leaf
SSLOCSPResponderCertificateFile /etc/pki/certs/responder.pem
SSLOCSPDefaultResponder "{OCSP Responder URL}"
SSLOCSPOverrideResponder on
```

Use case 4: Here OCSP Responder Signer Certificate is issued by the DIFFERENT (other then the Root) CA as CAC/PIV certificate; multi-Tier PKI (Root – Intermediate – Client), this OCSP Responder includes certificates for the entire PKI infrastructure.

For user case 4, OCSP configuration stanza will look like this:

```
SSLOCSPEnable on
SSLOCSPResponderCertificateFile /etc/pki/certs/responder.pem
SSLOCSPDefaultResponder "{OCSP Responder URL}"
SSLOCSPOverrideResponder on
```