NativeSPLogoutInitiator

The `<LogoutInitiator>` element is used to configure handlers that are responsible for initiating a logout operation, the termination of a user's session. The handler is responsible for performing protocol-specific tasks related to the logout, as well as terminating the session.

> ⊘ This is an advanced configuration feature. Most deployments can rely on the `<Logout>` shorthand element.

Logout can be "local" or "global". Local logout means that the SP's session is removed, but no communication with the IdP or other SPs is involved (with the caveat that the local logout might redirect to an IdP using some proprietary approach that is outside the scope of our documentation). Global logout implies that the IdP is also informed of the logout operation. The SP software includes user interface support for presenting a different template depending on which kind of logout takes place.

Initiation of logout via this mechanism can only be done by the user that owns the session (by contacting the handler while his/her session is active).

The ability to configure multiple LogoutInitiator handlers, and to combine them in chains, allows the deployer to control the selection of particular global /single logout protocols when more than one can be used, and to ensure that at least a local logout takes place.

Logout initiators are also required to invoke application notification loops during the logout operation. These are configured with the `<Notify>` element. Note that the actual `<Notify>` element(s) are configured at the application level, not inside the logout initiator.

Finally, you can supply a parameter named "return" as a query string parameter containing a URL. If logout completes successfully, the SP will redirect the browser to that location.

For some additional guidance, see the SLOWebappAdaptation topic.

Common Attributes

- `type`(string)
    - Plugin type name.

- `Location`(relative path)
    - The location of the LogoutInitiator (when combined with the base `handlerURL`). This is the location to redirect to when manually ending a session.

- `relayState`(string)
    - Controls how information associated with the logout request, primarily a URL to return to, is preserved for the completion of the logout process. Overrides the like-named attribute in the `<Sessions>` element.

- `signing` (see NativeSPSigningEncryption) (<u>Version 2.6 and Above</u>)
    - Controls outbound signing of XML messages subject to applicability to the protocol involved.

- `encryption` (see NativeSPSigningEncryption) (<u>Version 2.6 and Above</u>)
    - Controls outbound encryption of XML messages and content subject to applicability to the protocol involved.

SAML2 LogoutInitiator

Indicated by `type="SAML2"`, supports SAML 2.0 SP-initiated single logout. If the user's session was initiated with a protocol other than SAML 2, then the handler ignores the request. Otherwise, the initiating entityID is used to check for metadata with an `<md:IDPSSODescriptor>` role supporting SAML 2.0 and a compatible `<md:SingleLogoutService>` endpoint. The absence of either causes a warning to be logged and the handler otherwise ignores the request.

If a "return" query string parameter is provided, it will be preserved via a relay state mechanism.

Whether or not the logout request is successfully issued, the user's session will be removed if at all possible.

Attributes

- `template`(local pathname)
    - An HTML template used during transmission of the `<samlp:LogoutRequest>` message.

- `outgoingBindings`(space-delimited list of URIs)
    - List of SAML binding identifiers that determines the order of preferred `<md:SingleLogoutService>` bindings to use for the request. If this setting is used, failing to list a binding will prevent the use of an IdP that only supports the omitted binding.

- `postArtifact`(boolean) (defaults to false)
    - If true, the SAML artifact binding is implemented using a form POST rather then a redirect.

<u>Version 2.5 and Above</u>

- `asynchronous` (boolean) (defaults to true)
    - When true, the logout request will contain an extension signaling that the SP doesn't need a response back. This is used to simplify the typical use case in which the user interface is meant to stay at the IdP after the logout completes.

ADFS LogoutInitiator

⚠ The ADFS handler is only available if the `adfs.so` extension library is loaded by the SP.

Indicated by `type="ADFS"`, supports Microsoft ADFS "signout" requests. If the user's session was initiated with a protocol other than ADFS, then the handler ignores the request. Otherwise, the initiating entityID is used to check for metadata with an `<md:IDPSSODescriptor>` role supporting ADFS and a compatible `<md:SingleLogoutService>` endpoint. The absence of either causes a warning to be logged and the handler otherwise ignores the request.

A "supporting" IdP's role element has a `protocolSupportEnumeration` attribute containing the value `"http://schemas.xmlsoap.org/ws/2003 /07/secext"`, with an accompanying `<md:SingleLogoutService>` with a `Binding` of `"http://schemas.xmlsoap.org/ws/2003/07/secext"`.

If a "return" query string parameter is provided, it will be passed to the home realm STS in a "wreply" parameter.

Whether or not the logout request is successfully issued, the user's session will be removed if at all possible.

Local LogoutInitiator

Indicated by `type="Local"`, performs a local logout without attempting to involve the IdP, and removes the active session from the cache. Upon completion, the browser will be redirected to the location in the "return" query string parameter, if any, or the `localLogout` template will be displayed.

Chaining LogoutInitiator

Identified by `type="Chaining"`, wraps a sequence of LogoutInitiator handlers so that they run in series. The series ends when a handler indicates that a response to the browser was returned. If no response is sent, an error results.

Options specified via attributes on the surrounding element will apply to all the embedded handlers (if not overridden inside them).

Child Elements

- `<LogoutInitiator>`(one or more)
    - Embedded plugins to instantiate.