How to Configure Single Sign-On (SSO)

https://campus.barracuda.com/doc/4259872/

Single Sign-On (SSO) is a mechanism where a single set of user credentials is used for authentication and authorization to access multiple applications across different web servers and platforms, without having to re-authenticate.

The SSO system acts as a web gate for all inbound web traffic. When a user initially attempts to access the website, the user is challenged to provide login credentials. If authentication succeeds, an SSO user session cookie is generated. The SSO user session cookie authenticates the user for a period of time. If the log in fails, the authentication request is rejected.

The SSO environment protects defined resources (websites and applications) by requiring the following steps before granting access:

- **Authentication**: Authentication verifies the identity of a user using login credentials.
- **Authorization**: Authorization applies permissions to determine if this user may access the requested resource.

The Barracuda Web Application Firewall supports both single-domain and multi-domain SSO.

Single-Domain SSO

Single-domain SSO takes place within a single domain. For example, consider the domain `barracuda.com`, which hosts several restricted websites on several hosts. You can configure single sign-on for this domain, so that authenticated users can access all or a subset of the restricted resources by authenticating just once.

Set Up a Single-Domain Single Sign-On Environment

1. Go to the **ACCESS CONTROL > Authentication Policies** page.
2. In the **Authentication Policies** section, click the drop-down list under **Options** and select **Edit Authentication** for the service you want to configure Single Sign-On for.
3. In the **Edit Authentication Policy** window, do the following:
    1. Set the **Status** to *On*.
    2. Select the **Authentication Service** that needs to be associated with the service, and click **Show Advanced Settings**.
    3. In the **Session Control** section, specify the domain name of the services you are configuring Single domain SSO for in **Cookie Domain**. Example: service1 and service2 both have `. barracuda.com` as the session cookie domain.
    4. Specify values for the other parameters as required and click **Save**.

In a single-domain SSO set up, ensure that you configure the same **Session Cookie Domain** name for each service on the **ACCESS CONTROL > Authentication Policies** page.

Logging Out in a Single-Domain Single Sign-On Environment

When a user logs out of a domain, the Barracuda Web Application Firewall removes the user session cookie from the browser by expiring it, so the user is automatically logged out of other corresponding domains. For example, consider a user logged into `host1.bc.com`, `host2.bc.com` and `host3.bc.com` using `bc.com` as the cookie domain. If the user performs a logout in `host1.bc.com`, the user session cookie is removed from the browser, and the user is automatically logged out of `host2.bc.com` and `host3.bc.com`.

If the user does not access the SSO environment within the specified idle timeout, the user session becomes idle, and the user is challenged to provide login credentials to access the SSO environment again.

Multi-Domain SSO

Multi-domain SSO enables user authentication to be honored by hosts in two or more domains. For example, a set of URLs that reside within the domains `www.abc.com` and `www.xyz.com` can be set to single sign-on.

To achieve a multi-domain single sign-on, a parent domain is required for authentication. The Barracuda Web Application Firewall multi-domain single sign-on environment can have one parent domain and one or more child domains. The parent domain acts as a centralized authentication server that authenticates the users and transfers the SSO user session cookie to the child domains.

In a multi-domain single sign-on environment, each domain is responsible for maintaining and enforcing its own idle timeout. This means the cookie value for different domains might be different. You must configure the parent service and the child services on the Barracuda Web Application Firewall on the **ACCESS CONTROL > Authentication Policies** page.

Multi-Domain Single Sign-On Configuration

For a multi-domain SSO environment, specify the parent service and parent service URL for the domains as explained below:

- **Parent Service**: Specifies if the parent service URL is handled by this service. When set to *Yes*, this service acts as the parent domain to the subsequent domains. When set to *No*, this service

acts as the child domain that accepts the cookie from the parent domain.

- **Parent Service URL**: Specifies the URL that provides a cookie. In case of the parent domain, specify only the URL path. For the child domains, specify the protocol, host, parent domain, and URL path.

  - The parent service URL should be a virtual URL (internal URL). For example, `/ncsso.process`, `/index.html`, etc. This URL is used to identify the parent service URL in a multi-domain environment.
  - A global ACL rule must be created for the specified parent service URL on the **SECURITY POLICIES > Global ACL's** page with the following configuration settings:
    - The parameter **Action** must be set to *Allow*.
    - The configured parent service URL should be specified in the **URL Match** field.

For example, consider `www.abc.com` as the parent domain and `www.xyz.com` as the child domain. If the parent service URL for the parent domain is */ncsso.process*, then the parent service URL for the child domain is `http://www.abc.com/ncsso.process` .

Multi-Domain Single Sign-On Functionality

If a user attempts to access the parent domain first, the user is challenged to provide login credentials. On a successful login, the user gains access to the parent domain and to the child domains. But if a user attempts to visit the child domain first, the Barracuda Web Application Firewall redirects the user to the parent service URL for authentication, and challenges the user to provide login credentials. If successful, the user gains access and is redirected to the requested domain.

For example, consider `www.abc.com` as the parent domain and `www.xyz.com` as the child domain. If a user attempts to access the parent domain first (`www.abc.com`), the user is challenged to provide login credentials. An SSO user session cookie is generated on a successful login. Now, the user gains access and can navigate to the child domains using the generated session cookie without having to re-authenticate.

If a user attempts to access the child domain first (`www.xyz.com`), the web application redirects the user to the parent service URL for authentication. The user is challenged to provide login credentials. If successful, SSO user session cookies are generated for both domains (parent and child) and the user is allowed to access the child domain.

Set Up Multi-Domain Single Sign-On Environment

Configure Parent Domain

1. Go to the **ACCESS CONTROL > Authentication Policies** page.
2. In the **Authentication Policies** section, click the drop-down list under **Options** and select **Edit Authentication** for the service that needs to be configured as a parent domain.
3. In the **Edit Authentication Policy** window, do the following:

1. Set the **Status** to *On*.
2. Select the **Authentication Service** that needs to be associated with the service.
3. Scroll down to the **Single Sign On** section and set **Parent Service** to *Yes*. Note that this is the parent service that provides the cookie for the subsequent child domains.
4. Specify the URL path in the **Parent Service URL** field. The parent service URL path can be any URL you choose. Example: `/ncsso.process`, `/index.html`, etc. Note that a global ACL rule must be created for the specified parent service URL on the **SECURITY POLICIES > Global ACL's** page. For more information, see Multi-domain Single Sign-On Configuration.

4. Click **Save**.

Configure Child Domain

1. Go to the **ACCESS CONTROL > Authentication Policies** page.
2. In the **Authentication Policies** section, click the drop-down list under **Options** and select **Edit Authentication** for the service that needs to be configured as a child domain.
3. In the **Edit Authentication Policy** window, do the following:
   1. Set the **Status** to *On*.
   2. Select the **Authentication Service** that needs to be associated with the service.
   3. Scroll down to the **Single Sign On** section and set **Parent Service** to *No*. Note that this is the child service.
   4. Specify the protocol, host, parent domain and URL path in the **Parent Service URL** field.
   5. Click **Save**.

Chained Logout in a Multi-Domain Single Sign-On Session

If your logout is from the parent domain, the Barracuda Web Application Firewall removes the parent domain cookie from the browser by expiring it. If your logout is from the child domain, the child domain cookie is removed from the browser by expiring it, which redirects the user to the parent domain, and informs the parent domain to log out the user (remove the parent cookie).

For example, consider the case where three domains `www.xyz.com`, `www.abc.com` and `www.def.com` are a part of a multi-domain SSO environment, with `www.xyz.com` as the parent domain, and both `www.abc.com` and `www.def.com` as the child domains. When a logout is from the parent domain (`www.xyz.com`), the user session cookie is removed from the browser by expiring it, which automatically accomplishes the logout of the corresponding domains (`www.abc.com` and `www.def.com`).

When logout is from a child domain (e.g., `www.abc.com`), the child domain expires its cookie, redirects to the parent domain (`www.xyz.com`), and requests the parent domain to expire its cookie and log out. Then `www.abc.com` redirects the user to log out from `www.def.com`. To achieve this, configure **Auth Logout Success URL** as `http://www.def.com/nclogin.submit?f_method=LOGOUT` for authentication of `www.abc.com`.

In this case, the **Auth Logout Success URL** in `www.abc.com` is `http://www.def.com/nclogin.submit?f_method=LOGOUT` . This assumes that 'nclogin.submit' is configured as the **login-processor-path** in `www.def.com`. Similarly, for multiple child domains, you need to configure the same settings in `www.def.com` for the corresponding next domain and so on.

Steps Involved in Chained Logout

1. User performs a logout on `www.abc.com`. `www.abc.com` expires its cookie, and redirects the user to `www.xyz.com`.
2. `www.xyz.com` expires its cookie and redirects the user back to `www.abc.com`
3. `www.abc.com` redirects the user to perform a logout on `www.def.com`
4. `www.def.com` expires its cookie and redirects the user to `www.xyz.com`
5. `www.xyz.com` simply redirects the user back to `www.def.com`, since its cookie has been expired in Step 2.
6. The SSO user session cookie of all the three domains has been removed from the browser.
7. This process can be extended for more child domains by simply chaining the **logout-success-url** configuration in the authentication container.

If the user does not access the SSO environment within the specified idle timeout, the session becomes idle, and the user is challenged to provide login credentials to access the SSO environment again.