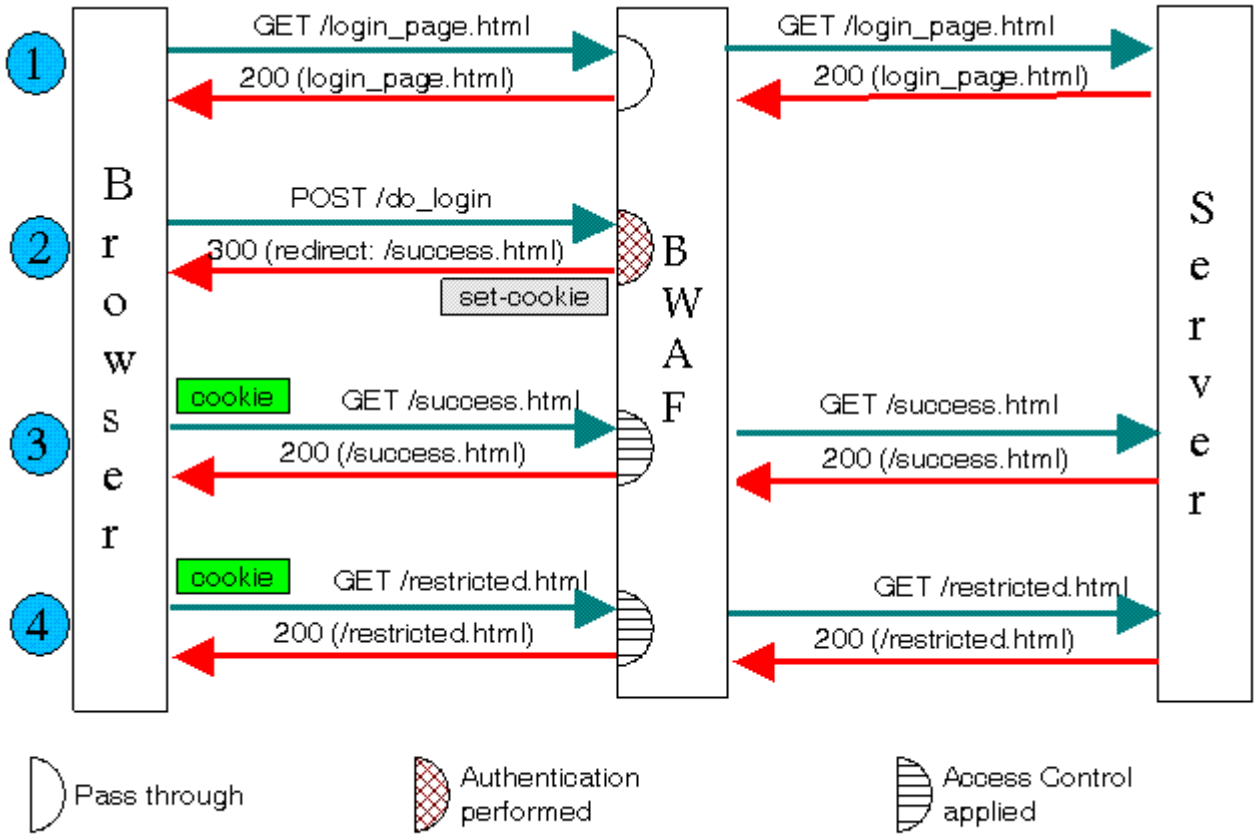
How to Configure Authentication and Access Control (AAA)

Overview

The Barracuda Web Application Firewall provides features to implement user authentication and access control. You can create a virtual private network (VPN) tunnel to control user access to websites. The user-access features allow you to specify who can access your websites and what access privileges each user has. By combining these with SSL encryption, you can create a secure VPN tunnel to your websites.

Authentication can be implemented only for HTTP or HTTPS services. The authentication process requires users to provide a valid name and password to gain access. A validated user has qualified access to the website; that is, the data and services this user can access depend on his authorization privileges. The following figure illustrates the authentication process:



The user accesses a login page (a GET request), a form for entering a username and password. The login form must be accessible to all users, but need not reside on a back-end server. The Barracuda Web Application Firewall includes a default login form which can be used instead of creating your own login page. The user submits the form (a POST request) and the Barracuda Web Application Firewall compares the submitted information against an internally or externally located authentication database. If successfully authenticated, the requester receives a cookie and is redirected to a success page. On subsequent requests, after verifying proper authorization (the authenticated user has the needed access privileges for the request) , the Barracuda Web Application Firewall forwards the request to the desired location.

If a user fails authentication, the user is redirected to a failed authorization page (not illustrated in the figure).

When an authenticated user attempts to access an unauthorized page, for which he does not have permission, he is redirected to a denied authorization page.

Steps to Configure Access Control

To configure access control to your website, do the following:

1. Configure an authentication database.
2. Associate the authentication database with your website.
3. Configure the authorization policy for your website.

Step 1 - Configuring an Authentication Database

An authentication database can be internal or external. To configure an internal database, set up local users and groups on the **ACCESS CONTROL > Local User/Group** page. To configure an external database, use the **ACCESS CONTROL > Authentication Services** page.

Configuring Internal Authentication Database

The Barracuda Web Application Firewall maintains an internal LDAP authentication database, to which the administrator is required to create users and groups using the **ACCESS CONTROL > Local Users/Groups** page. One or more users can be added to each group, and one user can belong to multiple groups.

To Create a Group

1. Go to the **ACCESS CONTROL > Local Users/Groups** page. In the **Groups** section enter a name for the group in the **New Group Name** field.
2. Click **Add**. The new group gets listed under **Available Groups**.

To Create a User

1. Go to the **ACCESS CONTROL > Local Users/Groups** page. In the **Users** section, specify values for the following:
    1. **New User Name** – Enter a name for the user. Note that this is the username used to authenticate the user.
    2. **Password** – Enter a password for the user.
    3. **User Groups** – Select a group name from the list of groups and click **Add**. If you wish to associate the user to multiple groups, perform the same step again.
2. Click **Add**.

Configuring External Authentication Database

External authentication databases are configured on the **ACCESS CONTROL > Authentication Services** page. The Barracuda Web Application Firewall supports the following authentication database servers:

- LDAP
- RADIUS
- SITEMINDER
- RSA SECURID
- SAML Identity Provider

Configuring LDAP Database Server

LDAP Authentication service identifies a database server supporting the LDAP protocol, which contains a set Authentication service. It is a unique identifier that identifies a set of users, groups, and contains mapping between the groups and the users. Configuration of this page allows the Barracuda Web Application Firewall to communicate with an existing LDAP directory server, and authenticate a user.

To enable LDAP user authentication:

1. From the **ACCESS CONTROL > Authentication Services** page select the **LDAP** tab.
2. Enter information about your LDAP server:
    1. **Realm Name** – Enter the name of the realm under which the Barracuda Web Application Firewall admins are stored (A realm identifies a collection of users and groups. It specifies information, in a flat directory structure, such as where users are located and where groups are located.).
    2. **Server IP** – Enter the IP address of the external LDAP server to authenticate users.
    3. **Server Port** – Enter the port number on which the server listens to LDAP connections. The standard LDAP ports are: port 389 for non-SSL connections and 636 for SSL connections.
    4. **Secure Connection Type** – Select the type of secure connection to be used by Barracuda Web Application Firewall when querying the LDAP database for user authentication and role retrieval.
        - **None** – Establishes a plain text connection.
        - **SSL/TLS** – With SSL you can create a SSL socket and send/ receive LDAP messages over it. Typically LDAP server accepts SSL connections on port 636. The LDAP URI for this is defined as ldaps://
          Transport Layer Security (TLS) protocol enables client/server applications to establish a secure connection over the Internet. TLS allows client/server applications to communicate in a way that is designed to prevent tampering or message forgery.
        - **StartTLS** – Upgrades an existing insecure plain text connection by sending an extended request to encrypt the connection using TLS.
3. Enter information about a user in your LDAP directory that has read access to all the users in LDAP directory:
    1. **Bind DN** – Enter Distinguished Name (DN) of the user to query the LDAP server.
    2. **Base DN** – Enter DN at which to start the search in the LDAP directory.
    3. **Bind Password** – Enter the password for the user querying the LDAP server.
    4. **Login Attribute** – Enter the attributes of an LDAP object used for identifying the user. For example: uid, sAMAccountName.
    5. **Group Name Attribute** – Enter the attributes of an LDAP object used for identifying the name of a group. Example: cn, sAMAccountName.
    6. **Group Filter** – Enter the filter attribute to retrieve the list of groups of the user in the LDAP directory. The maximum allowable characters are 500.
    7. **Query For Group** – Select *Yes* to query the groups in the LDAP directory for authentication. If set to *No*, queries are directed to individual user names for authentication.
4. Optional. Test the entered values for connectivity, username binding, and encryption:
    1. Click **Test LDAP**. The Barracuda Web Application Firewall checks the information you provided.
    2. Check the test results displayed at the bottom of the page.
    3. If the test fails, you can either correct settings as needed and repeat **Step 4**, OR you can use the **LDAP Discovery** tool as described in the next step.
5. Test the entered values and view troubleshooting details and recommendations (if any):
    1. Click **LDAP Discovery**. The Barracuda Web Application Firewall checks the information you provided.
    2. Check the test results:
    - Verified information is indicated with a green dot next to the field.
    - Information that need to be corrected is indicated with a red dot next to the field. **Note:** If you want to view detailed query results, click Verbose.
    - If any information is incorrect or missing, edit fields as needed and then repeat Step 5.
6. After your settings have been validated, click **Add** to save your settings.

Configuring RADIUS Database Server

The RADIUS protocol is based on a client/server model. The Barracuda Web Application Firewall can operate as a client of a RADIUS server. The client is responsible for passing user information to a designated RADIUS server and then acting on the response that is returned.

A RADIUS server (or daemon) can provide authentication and accounting services to one or more Barracuda Web Application Firewall devices. RADIUS servers are responsible for receiving user connection requests, authenticating users, and then returning all configuration information necessary for the client to deliver service to the users. A RADIUS server is generally a dedicated workstation connected to the network.

RADIUS Authentication service identifies a database server supporting the RADIUS protocol that contains a set of users, groups, and mapping between groups and users. This container allows the user to configure the Barracuda Web Application Firewall to communicate to an existing RADIUS directory server to authenticate a user.

To enable RADIUS user authentication:

1. From the **ACCESS CONTROL > Authentication Services** page select the **RADIUS** tab.
2. Enter information about your RADIUS server:
    1. **Realm Name** – Enter the name of a realm. A realm is a RADIUS compliant database of authorized user and group records. The realm can be located internally or externally on a RADIUS server.
    2. **Server IP** – Enter the IP address of the RADIUS server to authenticate users.
    3. **Server Port** – Enter the port number of the RADIUS server. Port 1812 is normally used for RADIUS.
    4. **Shared Secret** – Enter the secret key which is shared between the Barracuda Web Application Firewall and RADIUS server. Minimum value of the key is 6.
    5. **Timeout** – Enter the time in seconds for Barracuda Web Application Firewall to wait for a response from the RADIUS server before retransmitting the packet.
    6. **Retries** – Enter the number of times you want the Barracuda Web Application Firewall to transmit a request packet to the RADIUS server before giving up.
3. Click **Add** to add your RADIUS server configuration.

Configuring a Secondary RADIUS Server

The Barracuda Web Application Firewall supports secondary RADIUS server for authenticating users. In case the primary RADIUS server fails, the secondary RADIUS server takes over as the primary RADIUS server for authenticating users. When configuring the secondary server, note all parameter values including shared secret of the secondary RADIUS server must be identical to the primary RADIUS server, except the server IP address and port number.

To configure a secondary RADIUS server

1. Click **Add** next to the RADIUS authentication service for which you want to add the secondary server. The Authentication Services window appears, specify values for the following:
    1. **Secondary Server IP** – Specifies the IP address of the secondary RADIUS server.
    2. **Secondary Server Port** – Specifies the port number of the secondary RADIUS server.
2. Click **Add** to add the secondary RADIUS server configuration

Configuring SITEMINDER Database Server

SiteMinder Authentication service identifies a database server supporting the SiteMinder protocol, which contains a set of users, groups, and mapping between groups and users. This container allows the user to configure the Barracuda Web Application Firewall to communicate to an existing SiteMinder directory server for authenticating a user.

Support for SiteMinder has been deprecated. Also, SiteMinder feature will NOT be available from Version 9.1.

To enable SITEMINDER user authentication

1. From the **ACCESS CONTROL > Authentication Services** page select the **SITEMINDER** tab.

2. Enter information about your SITEMINDER server:
    1. **Realm Name** – Enter the name of the realm under which the Barracuda Web Application Firewall admins are stored.
    2. **Server IP** – Enter the IP address of the SiteMinder Policy Server to authenticate users.
    3. **Port** – Enter the authentication port of the SiteMinder Policy Server. Port 44443 is the standard port used for SiteMinder.
    4. Enter information about a user in your SITEMINDER server that has privilege to access all the SITEMINDER policies in SITEMINDER server.
        1. **Admin** – Enter the privileged username for the SiteMinder Policy Server.
        2. **Password** – Enter the privileged user's password for the SiteMinder Policy Server.
        3. **Agent Name** – Specifies the agent name configured in the SiteMinder Policy Server to represent Barracuda Web Application Firewall as SiteMinder agent.

           The specified agent name must have the following parameters set to *Yes* under **Agent Conf Objects** on the SiteMinder Policy Server:

           - AcceptTPCookie

           - RequireCookies

3. **Host Conf Object** – Enter the corresponding Host Configuration Object defined on the SiteMinder Policy Server.
4. **Shared WAF IP Addresses** – Enter the IP address(es) of the Barracuda Web Application Firewall that share the user sessions. Each system specified here should set **Single User Session** to *Yes* on the **ACCESS CONTROL > Authentication** page to synchronize information between each other, and keep only one active session for a user. For example, consider you have two systems with the IP addresses 10.10.10.10 and 10.10.11.11. The system with IP address 10.10.10.10 should have 10.10.11.11 configured as **Shared WAF IP Addresses** and vise versa. Also, both systems should set **Single User Session** to *Yes* to synchronize information and keep only one active user session. **Note**: When configuring this parameter, all Barracuda Web Application Firewalls should have the same **Realm Name** and **Cookie Encryption Key**.
5. Click **Add** to add your SITEMINDER server configuration.

Configuring RSA SECURID Database Server

RSA SecurID authentication service uses the RSA Authentication Manager database to authenticate the identity of users based on two factors: the current code generated on the user's assigned RSA SecurID authenticator, and a secret memorized Personal Identification Number (PIN) before granting access to protected resources.

To enable RSA SECURID user authentication:

1. From the **ACCESS CONTROL > Authentication Services** page select the **RSA SECURID** tab.
2. Enter information about your RSA RADIUS server:
    1. **Realm Name** – Enter the name of the realm under which the Barracuda Web Application Firewall admins are stored.
    2. **Server IP** – Enter the IP address of the RSA RADIUS server to authenticate users.

       The RSA Authentication Manager server running RADIUS is termed as RSA RADIUS server in the Barracuda Web Application Firewall.

    3. **Server Port** – Enter the port number of the RSA RADIUS server. Port 1812 is the standard port for RADIUS.
    4. **Shared Secret** – Enter the secret key which is shared between the Barracuda Web Application

Firewall and the RSA RADIUS server. Minimum value of the key is 6.
5. **Timeout** – Enter the time in seconds for Barracuda Web Application Firewall to wait for a response from the RSA RADIUS server before retransmitting the packet.
6. **Retries** – Enter the number of times you want the Barracuda Web Application Firewall to transmit a request packet to the RSA RADIUS server before giving up.

3. Click **Add** to add your RSA RADIUS server configuration.

Step 2 - Associating the authentication database with your website

The **ACCESS CONTROL > Authentication** page allows you to specify the parameters and resources to bind a configured authentication database with your Service and configure authentication of users.

Steps to associate an authentication database server with your website:

1. From the **ACCESS CONTROL > Authentication** page identify the Service to which you want to bind an authentication database.
2. Click **Edit** next to that Service. The **Edit Authentication Policy** window appears.
3. In the **Edit Authentication Policy** section, set the Status to *On* and select the authentication database server from the **Authentication Service** drop-down list.
4. Specify values to other parameters as required and click **Save**.

When LDAP is selected as an authentication database server, the **Auth Password Expired URL** field is displayed. Specify the URL to redirect the user when authentication fails due to an expired password. The Barracuda Web Application Firewall identifies the password expiry of a user and redirects the user to the specified URL to reset the password. This feature is supported ONLY for LDAP authentication service (Active Directory). Note that the expired password on the OpenLDAP server is not detected by the Barracuda Web Application Firewall.

Step 3 - Configuring the authorization policy for your website

The **ACCESS CONTROL > Authorization** page allows you to configure custom access across your website allowing or denying users or groups access to specific services. Access control for a service is configured per URL and Host Match. Configure access control for a URL key of a service to restrict which users/groups can access that URL space. Customized access is configured by user and/or group.

Steps to configure an authorization policy for your website:

1. Go to the **ACCESS CONTROL > Authorization** page. In the **Add Authorization Policy** section specify values for the following:
    1. **Service** – Select the Service to which you want to configure access control.
    2. **Policy Name** – Enter a name for the authorization policy.
    3. **Status** – Set to *On* to apply this authorization policy to the Service.
    4. **URL Match** – Enter a URL to be matched to the URL in the request. The URL should start with a "/" and can have at most one " * " anywhere in the URL. For example, /netbanking.html, any request matching this URL is required to authenticate before accessing this page. A value of “/*” means that the access control rule (ACL) applies for all URLs in that domain.
    5. **Host Match** – Enter a host name to be matched against the host in the request. This can be either a specific host match or a wildcard host match with a single “* “anywhere in the host name. For example, *.example.com, any request matching this host is required to authenticate before accessing this page.
    6. **Extended Match** – Define an expression that consists of a combination of HTTP headers and/or query string parameters. This expression is used to match against special attributes in the HTTP headers or query string parameters in the requests.
    7. **Extended Match Sequence** – Enter a number to indicate the order in which the extended match

rule must be evaluated in the requests.

8. **Login Method** –Select the login method to be used for authenticating users.
9. **Use Persistent Cookie** - When set to **Yes**, the authentication cookies set in the browser by the Barracuda Web Application Firewall are valid for the time period specified in **Persistent Cookie Timeout**.
10. **Persistent Cookie Timeout** - Enter the time in minutes to keep the persistent cookie valid, after which the cookie expires.

2. Click **Add** to associate the authorization policy with the Service.
3. To enforce fine grained access control, click **Edit** next to the Authorization Policy. The **Edit Authorization Policy** window appears.
4. In the **Edit Authorization Policy** section, specify values for the following:

    1. **Allowed Users** – Enter the list of users allowed to access the URL.

       To access to the URL, the user must be included either in "Allowed Users" or "Allowed Groups". For example, all the users in the group-HR get access to this URL when group-HR is listed in "Allowed Groups" parameter. Now, user-1 who does not belong to group-HR also needs access to the same URL. To achieve this, specify user-1 in "Allowed Users". This setting enables user-1 and group-HR get access to this URL.

    2. **Allowed Groups** – Enter the list of allowed groups to access the URL.

    3. **Auth Not Done URL** – Enter the URL where a user who attempts to access a protected URL before being authenticated will be redirected. If the URL is not specified, the user is redirected to the default login page generated by the Barracuda Web Application Firewall. Use this to redirect the user to a customized page or an error page instead of the default login page.

       The redirect URLs need not reside in the same service. Also, these pages must be hosted outside the Barracuda Web Application Firewall, typically in the server of the application. The internal Barracuda Web Application Firewall pages cannot be customized.

    4. **Access Denied URL** – Enter the URL where an authenticated user who lacks required access privileges should be redirected.
    5. **Send Basic Authentication** – Select *Yes* to convert user credentials to HTTP Basic Authentication header, included in every request sent to the server. This is useful when **Login Method** is set to *HTML Form*, and when the server needs to know the user credentials. Two typical cases of the server needing to know the user credentials are:
        1. To implement single sign on. The server may require customization to process the Basic Authentication header, extract the user ID and password, and perform any authentication or authorization required by the Service so a user is not challenged to log in again.
        2. To personalize the home page, the server needs to know the user ID. **Note**: HTTP Basic Authentication Headers are sent in clear text, so it is not a secure means of exchanging user credentials. The user ID and password are visible in the data packets transmitted to the server. It is recommended that this option is used only when the traffic to the server is encrypted.
    6. **Send Domain in Basic Authentication** – If set to *Yes*, the domain information of the client is forwarded to the server along with the user credentials in the Basic Authentication Header. This is applicable only when **Send Basic Authentication** is set *Yes*.

5. To perform advanced settings, specify values for the following in the **Advanced** section:
    1. **Authorization Agent** – The Barracuda Web Application Firewall is set as a default authorization agent to authorize users accessing this Service. This value remains the same for Services that are associated with the LDAP, RADIUS and RSA SECURID authentication services. For SITEMINDER authentication service, select **SiteMinder** as an authorization agent from the drop-down list to authorize users accessing SiteMinder protected resources.

2. **Authorization Result Cache** – Select an option from the drop-down list to cache the authorization result for allowed and denied responses.
    - **No Cache** – Does not cache authorization results.
    - **Cache All** – Caches all allowed and denied authorization results.
    - **Cache Allows** – Caches only the allowed authorization results.
6. **Authorization Resource Depth** – Enter the number of levels in the URI path of the request to be used for caching the authorization entries.
7. **Ignore Query String** – Set to *Yes* if you wish to ignore query strings in the request while caching authorization entries.
8. Click **Save**.

Related Articles

- How to Set Up a Custom Login Page for Authentication

Figures