Enabling Clickjacking Protection for a Service

https://campus.barracuda.com/doc/20251309/

Clickjacking, also known as UI redressing and iframe overlay, is a malicious technique where a user is tricked into clicking on a button or link on a website using hidden clickable elements inside an invisible iframe. This attack hijacks clicks intended for the visible page and routes the user to an application and/or domain on another page. The Barracuda Web Application Firewall uses the X-Frame-Options HTTP response header to detect and prevent iframe based UI redressing. The X-Frame-Options header is inserted to indicate whether a browser should be allowed to render a page in an iframe, and if allowed, the iframe origin that needs to be matched. The three values of the X-Frame-Options header are:

- **Never** – The browser will not display the page if the page is within the iframe.
- **Same Origin** – The browser allows the page to be displayed if the page within the iframe is from the same origin.
- **Allowed Origin** – The browser allows the page specified in the **Allowed Origin** to be displayed when embedded in the iframe.

- When Clickjacking is enabled for a service, make sure the **NO Response Rewrite** rule is configured with the header name 'X-Frame-Options' for that service in the **WEBSITES > Website Translations > HTTP Response Rewrite** section. If the back-end server is inserting 'X-Frame Options' header in the response, then enabling Clickjacking or configuring the response rewrite rule is not needed.
- If your website is rendered inside an iframe, then Clickjacking should not be turned *On* as it will prevent rendering the website inside the iframe. By default, Clickjacking is turned *Off*.

To enable Clickjacking protection for a service:

1. Go to the **WEBSITES > Advanced Security** page.
2. In the **Clickjacking Protection** section, identify the service you want to enable clickjacking protection for, and click **Edit** next to it. The **Edit Clickjacking Protection** window appears.
    1. Set **Status** to *On*.
    2. Select the appropriate option next to **Render Page Inside Iframe** to specify how the page should be rendered in an iframe.
    3. If **Render Page Inside Iframe** is set to *Allowed Origin*, specify the page/URL in the **Allowed Origin URI** field that needs to be displayed when embedded in the iframe.
3. Click **Save**.