

**Password Vault: HaveIBeenPwned.com Integration**

Version Dated: May 2, 2018

HaveIBeenPwned.com provides a free API to check passwords against a database of over half a billion passwords that have been compromised by hackers. If a previously hacked password is part of a new hack, it can be compromised more quickly than a unique password.

Js-haveibeenpwned is a lightweight JavaScript library that provides easy integration into any web solution. It allows a password to be checked against the growing database of previously hacked passwords at HaveIBeenPwned.com. It is asynchronous and provides a callback for full response customization.

Visit <https://haveibeenpwned.com/API/v2> for more information about the HaveIBeenPwned API.

*Note: js-haveibeenpwned requires the js-sha1 library. This library is not included in this solution, and is available here:* [*https://github.com/emn178/js-sha1*](https://github.com/emn178/js-sha1)*).*

# General Use

To use js-haveibeenpwned, ensure that the js-haveibeenpwned.js and js-sha1.js JavaScript libraries have been included in the web solution.

To check a password:

//Ex1: get true/false response

var password = document.getElementById("password").value;

haveIBeenPwned(password, function (isPwned) {

    var msg = isPwned ? "A match was found" : "Safe password :-)";

    alert(msg);

});

//Ex2: get full response

haveIBeenPwned(password, function (isPwned, matchCount, responseText) {

    //isPwned      => true if a matching password found

    //matchCount   => The number of time this password showed up in

    //                the HaveIBeenPwned database. Higher

    //                count may indicate a weaker password.

    //                Returns -1 if there was an API error.

    //responseText => The full response text from the HIBP API.

});

# netFORUM Integration

While this solution provides the files to facilitate the API calls, the integration into an eWeb or CMS page will require custom implementation. The library can be included as-needed by adding a new web content detail to the password page. Also, a system option can be used to customize the message to the users if the password is weak.

Insert a HTML web content detail with the following code:

<script src="/eWeb/Scripts/js-sha1.js" type="text/javascript"></script>

<script src="/eWeb/Scripts/js-haveibeenpwned.js" type="text/javascript"></script>

<script type="text/javascript">

//set the system option value to a variable

window.HIBPSystemOptionMessage = "{UserAlertMessageSystemOption}";

//Any other client side handling which gets the password and

// checks it against HaveIBeenPwned using the examples above.

// The “HIBPSystemOptionMessage” variable will be globally available.

</script>

For netFORUM, it will likely be simpler to download the sha1 and HIBP libraries and combine them into a single JavaScript file along with the page-specific implementation code. Assuming that has been done, and the resulting JavaScript file is named “haveibeenpwned-bundle.js”, the web content detail code would look like this:

<script type="text/javascript">

//set the system option value to a variable

window.HIBPSystemOptionMessage = "{UserAlertMessageSystemOption}";

</script>

<script src="/eWeb/Scripts/haveibeenpwned-bundle.js" type="text/javascript"></script>

For an example implementation of this solution, visit www.AIAA.org and take steps to create a new account.