Cross Site Request Forgery (CSRF) implementation

With the release of Open-AudIT 2.2.2 we have improved the CSRF implementation.

You still receive a token with every request from Open-AudIT. You still need to submit that token with the HTML form to create an item. At this stage HTML forms **only** need to do this. Going forward, all POST, PUT, PATCH and DELETE requests will need to include the access token. This is NOT required for 2.2.2 though.

We have introduced two new configuration items.

access_token_enable - This is set to **y** by default and will enable the requirement to use access tokens. If set to **n**, the CRSF access token verification will be disabled.

access_token_count - Set to **20** by default. This means you can submit any of the last 20 access tokens for validation.

The access token is regenerated with every request.

In normal use, no difference should be seen by the user.

In certain circumstances, the CSRF protection will kick in and reject a submitted form. This should not happen often (if ever) and attempting a second time to create an item or logging off, then back on should resolve the issue.

You can see the access token in the JSON response meta -> access_token (as below).


```
{
  "meta":
    {
      "access_token":"42578d44ac02490137917797bd722acb4ff7d5bdf62a04ba4346553638c3",
      "action":"collection",
      "baseurl":"http......
```

OUT OF DATE as at Open-AudIT 2.2.2.

Open-AudIT 2.2.1 includes our initial implementation to mitigate CSRF.

Whenever you request a page from Open-AudIT, contained in the response is an access token.

Whenever you submit a POST form to create an item, the form will contain this access token.

The access token is regenerated with every request.

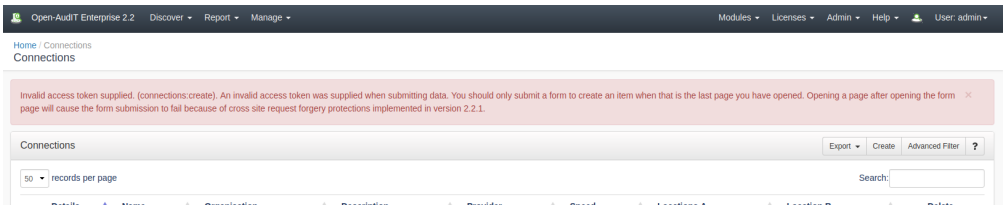
In normal use, no difference should be seen by the user.

In certain circumstances, the CSRF protection will kick in and reject a submitted form.

The most obvious example is if the user call the create widget (or anything else) form, then calls a second page in another tab, then finally submits the original form. This will be rejected as the token in the form will not match the token stored by Open-AudIT. The user will see the below error message.

Invalid access token supplied. (connections:create). An invalid access token was supplied when submitting data. You should only submit a form to create an item when that is the last page you have opened. Opening a page after opening the form page will cause the form submission to fail because of cross site request forgery protections implemented in version 2.2.1.

And the below screenshot (click to enlarge).



Another situation will occur where the following conditions are met:

In the Enterprise config file, auth_method_1 is **not** openaudit.

The user requests a create item form.

The task checker fires and completes before the user can submit the form.

The user submits the form.

Because the task checker is using the Open-AudIT Enterprise account AND the user is being validated by something other than Open-AudIT (and hence is also using the Open-AudIT Enterprise account), the CSRF prevention will run and the form will be rejected (again, with the same error as above).

In both cases, reloading the form and resubmitting should create the item.

As this is the initial implementation only, we have already devised improved logic to prevent both these situations from occurring, however we consider this important enough to release Open-AudIT 2.2.1 as it is.

This page will be updated with the improved logic details upon their inclusion in the code base.