Mozilla Security Review

General Information

Project Name:	Backend Localization Wordpress Plugin
Security Reviewer:	Matt Fuller (:mfuller)
Date of Review:	7/27/12
Bugzilla Bug #:	778244

Background Information:

We have a few blogs in non-English languages that need to be administrated by English speakers (currently blog.mozilla.org/laguaridadefirefox with more to come). To properly localize a blog we need to specify the language in a configuration file, which sets the language for the entire blog, both front-end and back-end. We also have English blogs whose authors may prefer to read the admin panel in their native language even if they're blogging in English. This plugin allows a blog's front-end to be a different language from its back-end, and lets the individual user select her/his preferred admin language.

Helpful URLs:

Plugin Page: http://wordpress.org/extend/plugins/kau-boys-backend-localization/

Access Information:

How is the application accessed?
Is the application internal or publicly available?
If it is internal, what mechanism prevents non-members from accessing it?
Are there links to user-only resources displayed to non-users?
Are login pages secured with SSL over HTTPS?
If HTTPS is used, is Strict Transport Security set?

The plugin is accessed via the admin control panel where a language can be set. The app is only accessible internally (after login) but a small part is visible to logged out users (the select language drop down on the login page).

Infrastructure and Backends:

What languages do the applications use?
What database language is used if applicable?
Are the running versions up to date?
What server is it running on?

PHP - WP Plugin

Accounts and Passwords:

If the mechanism to prevent general access is a password, how is the signup process handled?
How is account information stored?
Are passwords properly stored within databases if applicable?

Is a password policy in place?
Are accounts locked-out after a number of invalid logins?
Are passwords 8 characters or greater?
Do passwords use both numbers and letters (and possibly symbols)?
Is there a blacklist of common passwords?
Do passwords expire after X days and require a reset?
Are invalid logins logged?
Is there a lockout after X invalid attempts?
Is the error message for lockout generic (does not include if user/pass is valid)?
How are password resets handled (i.e. email, security question, etc.)?
Do emails sent after signup/reset contain a session link? (should not)
Do email verification codes expire after one use or 8 hours?
Is password reuse limited/prevented?

N/A - handled by Wordpress

Session Management:
How long are session cookies stored?
Are session tokens 128-bit or greater?
Is session ID token creation handled by the server (or cryptographically if locally)?
Do authenticated sessions timeout after 15 minutes?
Is the Secure Flag enabled for all set cookies?
Is the HTTP-Only flag used to disable malicious script access to the session ID?
Are new session ids created on login?
On logout, are session ids invalidated?

N/A - handled by Wordpress

Third-Party Resources:
Are third-party resources used (i.e. JavaScript libraries, images, CSS, etc.)?
Can those resources be trusted / are they from reputable sources?
Is there a chance the resource could be compromised?
Is it possible to host the resources locally to mitigate risks?
Is a third-party responsible for storage of user data?
Does the application connect with services like Facebook, Twitter, etc?

None.

Data Handling:
What kind of data is transferred between the user and the application?
Is this data generated by the user or generated automatically?
Can the data be trusted?
How is the data sent to the application (i.e. JSON format, plain-text, GET/POST, etc.)?
How is the data handled by the server/application?
Can the data be manipulated in transit?
What happens if the data is altered?
What is done with the data once it is received (i.e. stored in a database, displayed to users)?
Is any data storage done via cookies? If so, what kind of data is stored via this method?

The only data is a language selection. The data is passed via GET such as lang=en-US. This data is used to search for the correct language and use that file to display the admin

page in that language.

Uploaded Data:
Can the user upload data to the application?
Are extensions, file size, file type (not only based on extension), etc. checked?
Are files renamed upon upload?
Is the correct content-type used?

No uploads

Data Sensitivity:
What kind of data is being stored and/or manipulated by the application?
Does this data need to be encrypted in transit? In storage?
What is the impact if this data is lost/stolen?
Is secure/sensitive data sent over SSL?

N/A

Application Administration:
Is there an administration console?
Can it be accessed publicly?
How is it secured if so?
Are correct methods used to prevent admin actions from being performed outside of the admin console (i.e. using CSRF tokens)?
Are there any configuration pages that should not be made public?

WP Admin page - handled by Wordpress

Security Coding Flaws:
Have all user inputs been sanitized?
Is a maximum size for data (input or uploads) defined?
Do all URL variables pass through sanitization?
Is data from databases escaped properly?
Are CSRF tokens used to prevent POSTs from outside websites?
If a database is used, are protections against SQL injection in place?
Is validation done on the server side (not just client-side)?
Is output encoded in addition to sanitization of input?
Does the user ever send data to the OS level?
Are x-frame options sent to “deny” or “sameorigin”?
Is debug mode disabled?

Not all inputs sanitized
CSRF tokens not used, but only damage is changing language.

http://172.16.242.132/wp-admin/index.php?kau-boys_backend_localization_language=%22%3E--%3E%3Cscript%3Ealert%281%29;%3C/script%3E

Testing:

List all tests performed on the application

1. Reviewed code
2. Check for echo without encoding
3. Check for XSS
4. Check for CSRF
5. Look for other flaws

Results:

List the results of the above tests

1. No major issues (besides below)
2. Only in comments (lang=)
3. XSS Found
4. CSRF found, but no major issue
5. No major issues

Meetings and Notes: