HttpAuthenticationAdmin

General Description

A common way of protecting webpages is through Basic HTTP authentication. The web server sends a "401 Authentication Required" header when a protected page is requested. The browser would then prompt the user for a username and password. Access is allowed if the username password pair are valid; else, the web server sends a HTTP 401 error, meaning "access denied." HTTP authentication is usually used by creating a .htaccess file. (Only in Apache?)

Tikiwiki is able to detect when a visitor to the site is currently logged in using Basic HTTP Authentication. If the username of the user matches a username within Tikiwiki's database, Tikiwiki will automatically log the user in and, of course, grant all the assigned permissions.

Key Function and sub-features

HTTP Authentication is a very simple way to protect any kind of web content, content outside of TikiWiki's permission system. With this feature, servers with many different sections of content can all use one single login through HTTP authentication; users don't have to remember a separate Tikiwiki username password.

Related Links

http://httpd.apache.org/docs/howto/auth.html
http://us2.php.net/features.http-auth

Typical Uses

Integrating Tikiwiki into a broader authentication system

Case Studies

Central Authentication Service is a Web Initial Sign-on (WebISO) system designed by Yale ITS. CAS facilitates single sign-on across multiple web applications and provides these web services with the ability to authenticate users without having access to their passwords. From an end-user point of view, all protected pages show a standized CAS challenge page where the user types in their NetID (a unique username of sorts assigned to everyone affiliated with Yale) and password.

Much to our delight, we were able to make Tikiwiki interface with CAS without any customization. Yale ITS provides mod_cas, an Apache modules that protect webpages through CAS. Since mod_cas is an Apache module, it behaves like standard HTTP authentication. Tikiwiki supports HTTP authentication. When a user is logged in through HTTP authentication, and the username matches one of the usernames in the Tikiwiki database, Tikiwiki automatically logs the user in. That way, when a user logs in through CAS, Tikiwiki matches the NetID (username) of the user with a pre-created account in its database, and logs the user in.

(Taken from the YaleUniversityITS Case Study.)

Bugs

When using HTTP Authentication, there is no official way to log out other than closing the browser and ending the session. This is a flaw in Basic HTTP Authentication and not in TikiWiki. Here are some tricks/workarounds that may be used to simulate logout:

http://www.ssi-developer.net/htaccess/htaccess_logout.shtml

Support Requests

This doesn't appear to work - try for example going to http://foo:bar@tikiwiki.org/tiki-view_articles.php where foo and bar are your Tikiwiki login id and password, although its possible that tikiwiki.org is not setup to use http authentication. When I go Admin->Login->Authentication Method = Web Server, (and set http host to my server name e.g. natcap.mitra.biz then it refuses URLs of the form http://mitra:xxxx@natcap.mitra.biz treating me as an unregistered user. Is this all that is required to configure this, or is there something non-obvious. (mitra at mitra.biz)

Admin Documentation

In Admin / Login (tiki-admin.php?page=login), change **Authentication method** to *Web server*. Create user accounts for appropriate users who will use HTTP authentication. The BatchUserImport feature is very useful here. Now, when users access Tikiwiki while logged on using HTTP Authentication, and their username matches one in the Tikiwiki user database, they will automatically be logged into TikiWiki.

Notes

This is a new feature in v1.7.

Knowledgebase / tutorial / FAQ / How-to

For more information