

Enable Apache Auth in op5 Monitor

It's tricky, it's by design messy, and I don't encourage people to use it if possible not to.

Redirection Notice

This page will redirect to [DOC:Apache](#) in about 3 seconds.

First of all, apache driver can't handle groups. That's, as far as I know, and what I can read from the source, on answers on internet, and from the not-existing documentation, it's actually impossible to get groups out of apache auth/n//z/-modules to PHP. There isn't any infrastructures for it at all. Groups only exist within each auth module in apache. That means, using apache modules, you loose all benefits of using groups, and set permissions on groups.

With that said, enabling apache isn't as easy as enabling an LDAP driver.

Prepare the auth modules:

1. Go to nacoma and "Auth Modules"
2. Create a new driver of type "Apache", name isn't important, call it Apache
3. go to common settings for auth modules, and check "Enable Auto Login" (which is an awful name, it tells the auth drivers to every request try to handle the login operation, which is needed for apache auth). If this isn't available, as in some 6.0 versions, you need to manually enable it in `/etc/op5/auth.yml` and the field `"enable_auto_login: true"` in common section

Prepare the groups:

4. Go to "Assign group rights"
5. Lookup a user that should be available through apache driver later, though "Lookup user". You should see some groups: `apache_auth_user`, then there is also a group `user_yourusername`, which is available for all users, so rights can be assigned to individual users.
6. Assign rights, and make sure that someone have access to configure the system when apache auth is configured later.

Now monitor should be prepared to use apache auth. Setup Apache authentication and authorization as wanted, and enable it for the following paths:

`/ninja`

`/monitor`

`/monitor/op5/nacoma` (beware, if using `.htaccess`, this is using a separate Alias directive, but nacoma shouldn't work at all without it)

`/monitor.old`

The auth driver just listens to the authenticated username, so enabling this driver, giving permissions, but no auth-handling in apache would open the system to all users forcing an authentication header.