Cornell University

Authentication / Authorization with Drupal

Topics:

- Review Authentication / Authorization
- Drupal Modules
  - SimpleSAMLphp_auth + SimpleSAMLphp (Jeff)
  - webserver_auth (Steve)
  - LDAP (Eric)
- Q/A

Panel:

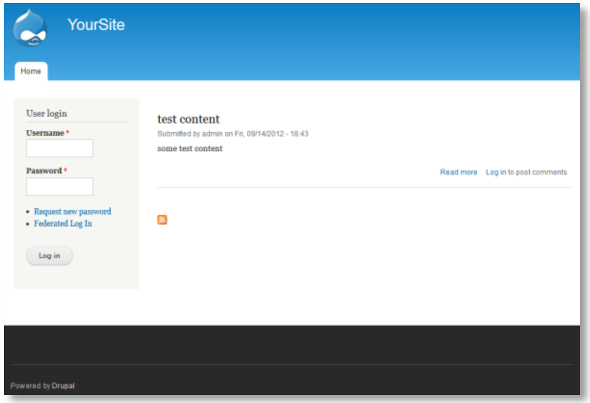
- Jeff Amaral, Developer, Singlebrook Technology
- Eric Chen, Systems Engineer, CIT Infrastructure
- Steve Gaarder, Systems Administrator, Dept of Mathematics

Authentication / Authorization with Drupal

Authentication: “Local”

Username/password

“OK”



test content

Submitted by admin on Fri, 09/14/2012 - 18:43

some test content

Pros

- Works out of the box

Cons

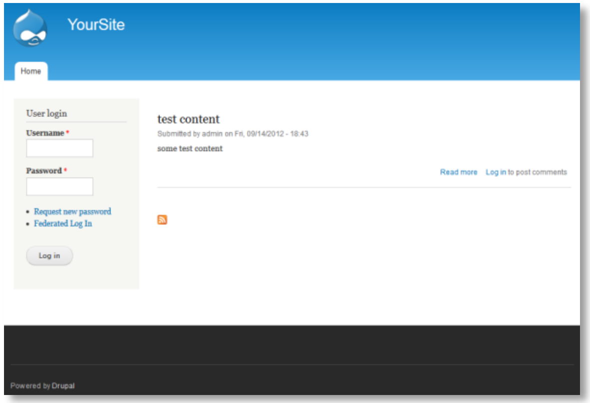
- Separate credential
- Need to worry about storing password ☹

Authentication / Authorization with Drupal

Authentication: “Proxy” (NOT RECOMMENDED)

Username/password →

← “OK”



test content

Submitted by admin on Fri, 09/14/2012 - 18:43

some test content

Username/password ↘

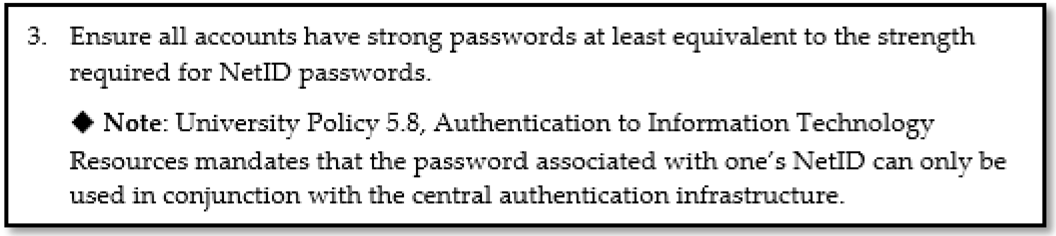
CornellAD

Pros

- Integrates with external authentication

Cons

- Violates Cornell Policy 5.10 (for NetIDs)



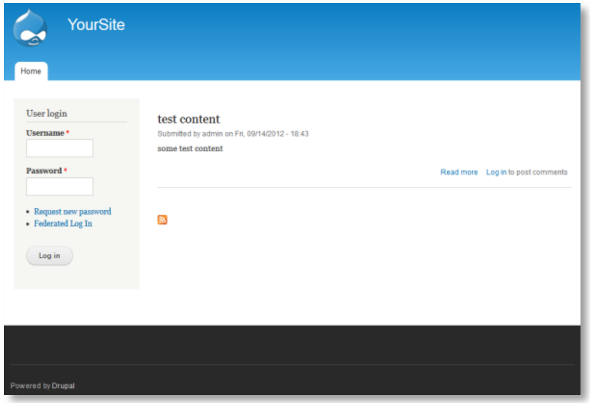
3. Ensure all accounts have strong passwords at least equivalent to the strength required for NetID passwords.

   ◆ Note: University Policy 5.8, Authentication to Information Technology Resources mandates that the password associated with one’s NetID can only be used in conjunction with the central authentication infrastructure.

Authentication / Authorization with Drupal

Authentication: SimpleSAMLphp

“OK”



YourSite

Home

User login

Username *

Password *

- Request new password
- Federated Log In

Log in

test content

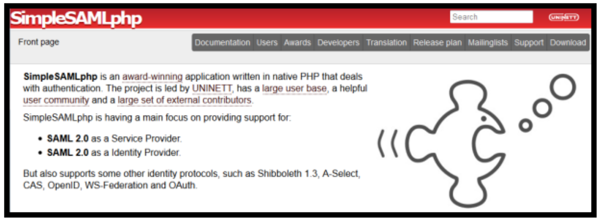
Submitted by admin on Fri, 09/14/2012 - 18:43

some test content

Read more Log in to post comments

Powered by Drupal

“OK”



SimpleSAMLphp

Search

Front page

Documentation | Users | Awards | Developers | Translation | Release plan | Mailinglists | Support | Download

SimpleSAMLphp is an award-winning application written in native PHP that deals with authentication. The project is led by UNINETT, has a large user base, a helpful user community and a large set of external contributors.

SimpleSAMLphp is having a main focus on providing support for:

- **SAML 2.0** as a Service Provider.
- **SAML 2.0** as a Identity Provider.

But also supports some other identity protocols, such as Shibboleth 1.3, A-Select, CAS, OpenID, WS-Federation and OAuth.

“OK”

Username/password

CU WebAuth

Pros

- Uses Campus SSO
- Only requires PHP

Cons

- Does not protect non-code resources
- Some complexity to configure
- Does not handle Weill Cornell IDs

Authentication / Authorization with Drupal

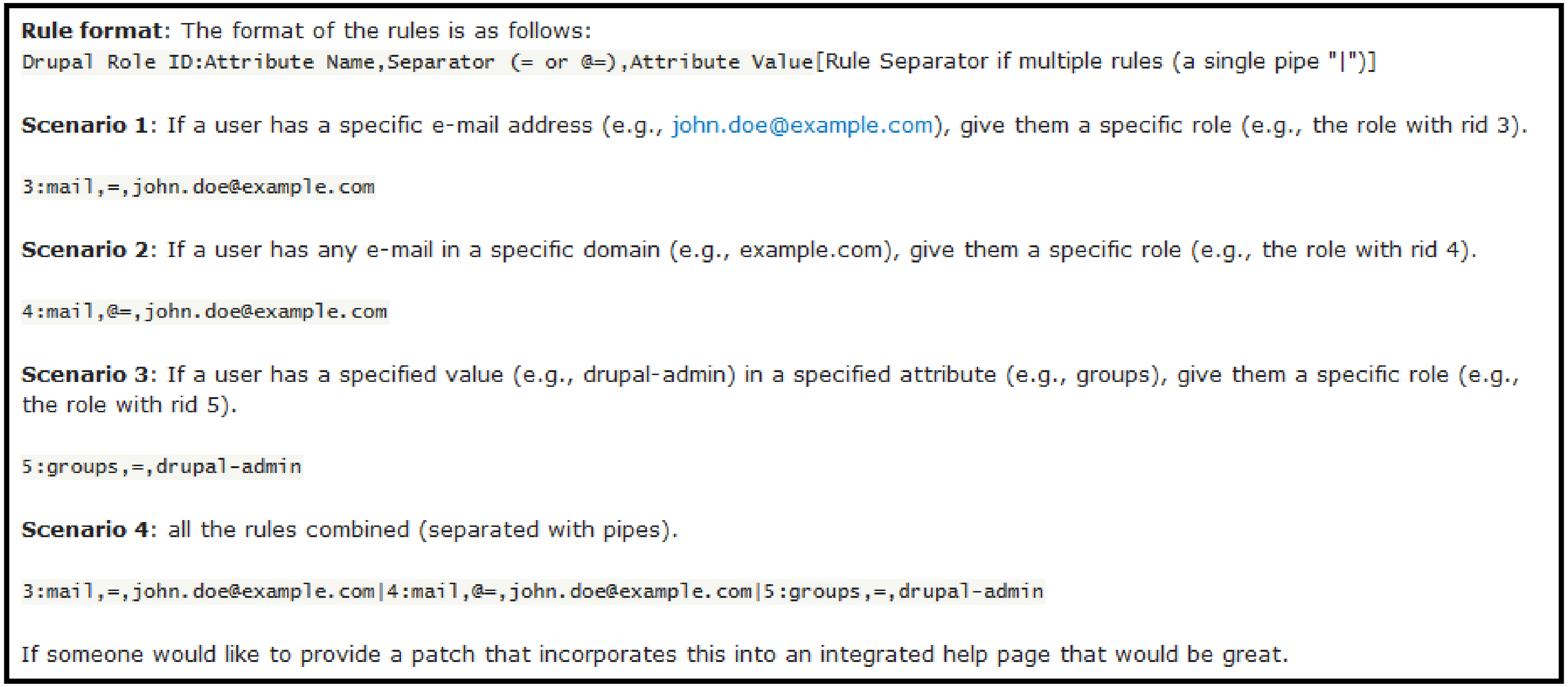
Authorization: SimpleSAMLphp

How to: https://confluence.cornell.edu/x/igEkD

Authentication / Authorization with Drupal

Authorization: SimpleSAMLphp

http://drupal.org/node/1931394



Rule format: The format of the rules is as follows:
`Drupal Role ID:Attribute Name,Separator (= or @=),Attribute Value`[Rule Separator if multiple rules (a single pipe "|")]

Scenario 1: If a user has a specific e-mail address (e.g., john.doe@example.com), give them a specific role (e.g., the role with rid 3).

```
3:mail,=,john.doe@example.com
```

Scenario 2: If a user has any e-mail in a specific domain (e.g., example.com), give them a specific role (e.g., the role with rid 4).

```
4:mail,@=,john.doe@example.com
```

Scenario 3: If a user has a specified value (e.g., drupal-admin) in a specified attribute (e.g., groups), give them a specific role (e.g., the role with rid 5).

```
5:groups,=,drupal-admin
```

Scenario 4: all the rules combined (separated with pipes).

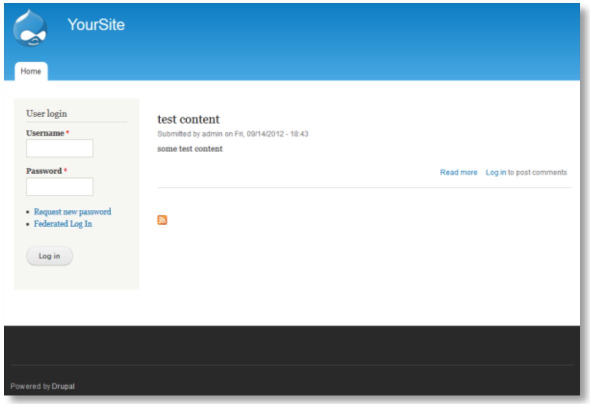
```
3:mail,=,john.doe@example.com|4:mail,@=,john.doe@example.com|5:groups,=,drupal-admin
```

If someone would like to provide a patch that incorporates this into an integrated help page that would be great.

Authentication / Authorization with Drupal

Authentication: CU WebAuth

"OK"



YourSite

Home

User login

Username *

Password *

- Request new password
- Federated Log In

Log in

test content

Submitted by admin on Fri, 09/14/2012 - 18:43

some test content

Read more Log in to post comments

Powered by Drupal

Username/password

"OK"

CU WebAuth

Pros

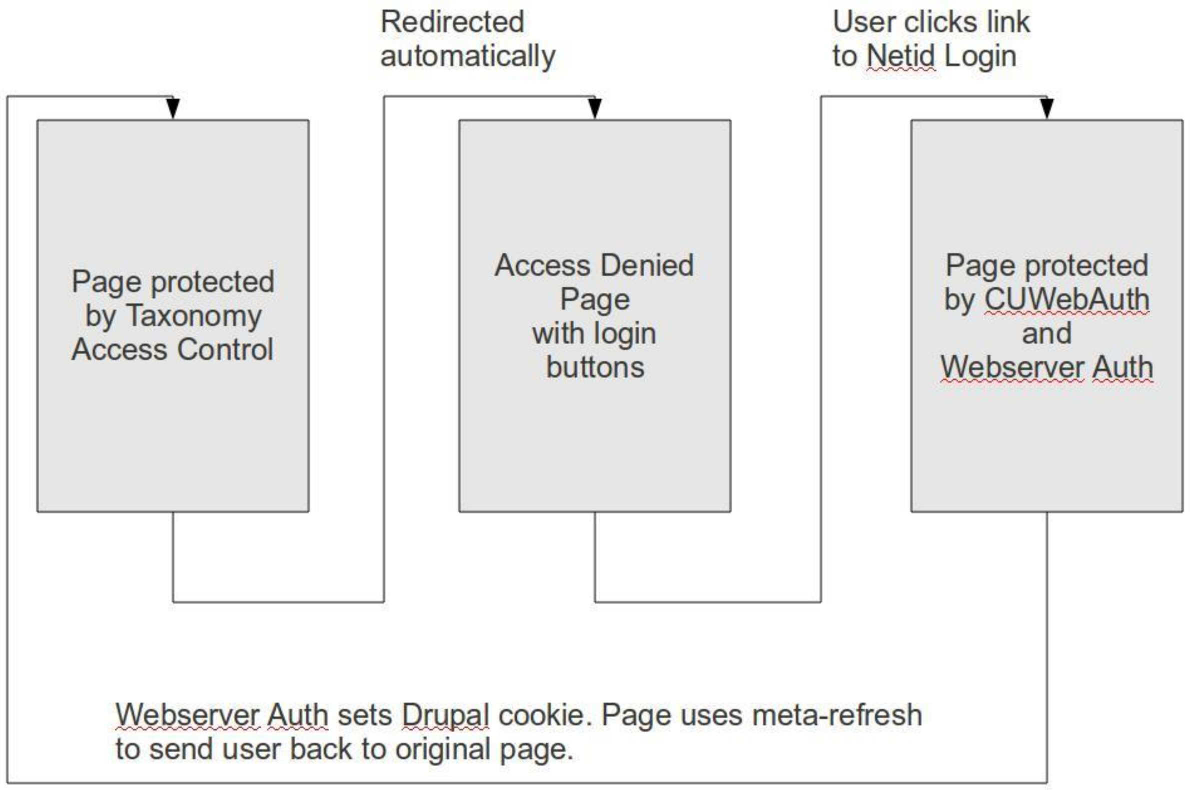
- Uses Campus SSO

Cons

- Requires Apache module

Authentication / Authorization with Drupal

Authentication: CU WebAuth



Redirected automatically

User clicks link to Netid Login

Page protected by Taxonomy Access Control

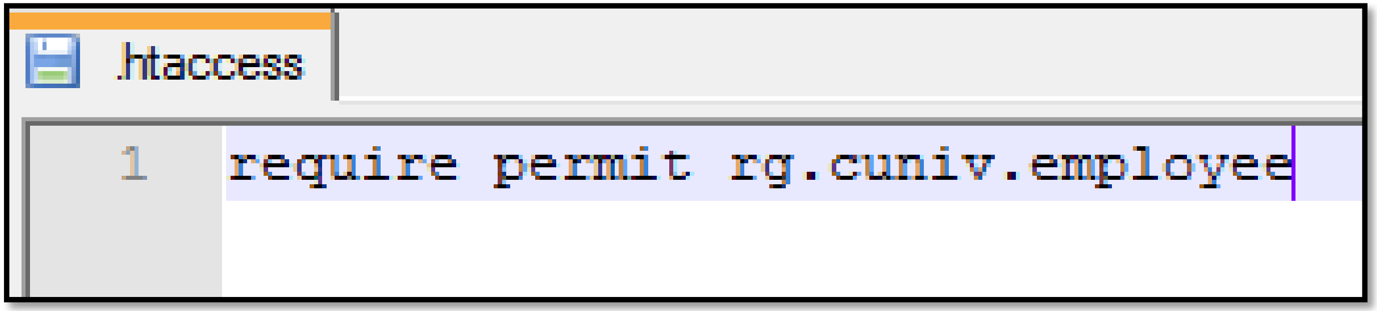
Access Denied Page with login buttons

Page protected by CUWebAuth and Webserver Auth

Webserver Auth sets Drupal cookie. Page uses meta-refresh to send user back to original page.

Authentication / Authorization with Drupal

Authorization: CU WebAuth



```
require permit rg.cuniv.employee
```

Restricts all access to Cornell Employees

Restriction is enforced by web server (Apache HTTPD or MS IIS)

Pros

- Uses CornellAD
- Can protect non-code resources

Cons

- Applies a “blanket” to entire site

Authentication / Authorization with Drupal

Authorization

- Previous examples are suitable to on-campus/off-campus/cloud
- LDAP examples will only work from on-campus

Authentication / Authorization with Drupal

Authorization: Drupal LDAP

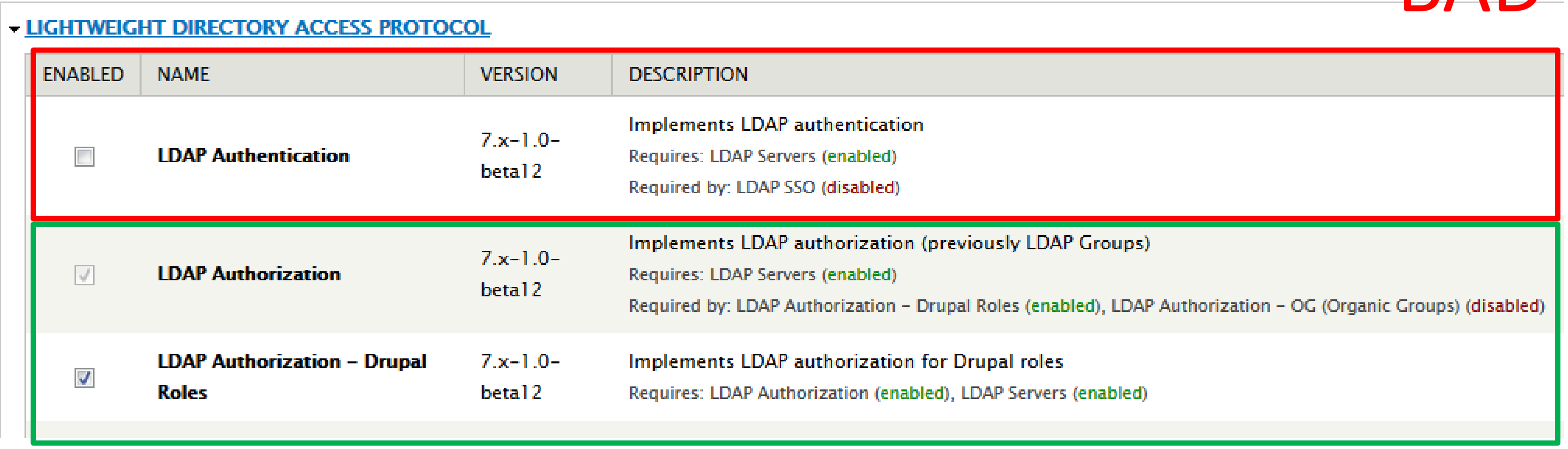
- Requires a CornellAD HoldingID (contact Identity Management)

Authentication / Authorization with Drupal

Authorization: Drupal LDAP

“LDAP Authentication” is an example of “pass through” authentication and is NOT RECOMMENDED

BAD



▾ LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL

ENABLED	NAME	VERSION	DESCRIPTION
□	**LDAP Authentication**	7.x-1.0-beta12	Implements LDAP authentication Requires: LDAP Servers (enabled) Required by: LDAP SSO (disabled)
☑	**LDAP Authorization**	7.x-1.0-beta12	Implements LDAP authorization (previously LDAP Groups) Requires: LDAP Servers (enabled) Required by: LDAP Authorization – Drupal Roles (enabled), LDAP Authorization – OG (Organic Groups) (disabled)
☑	**LDAP Authorization – Drupal Roles**	7.x-1.0-beta12	Implements LDAP authorization for Drupal roles Requires: LDAP Authorization (enabled), LDAP Servers (enabled)

Good

Pros

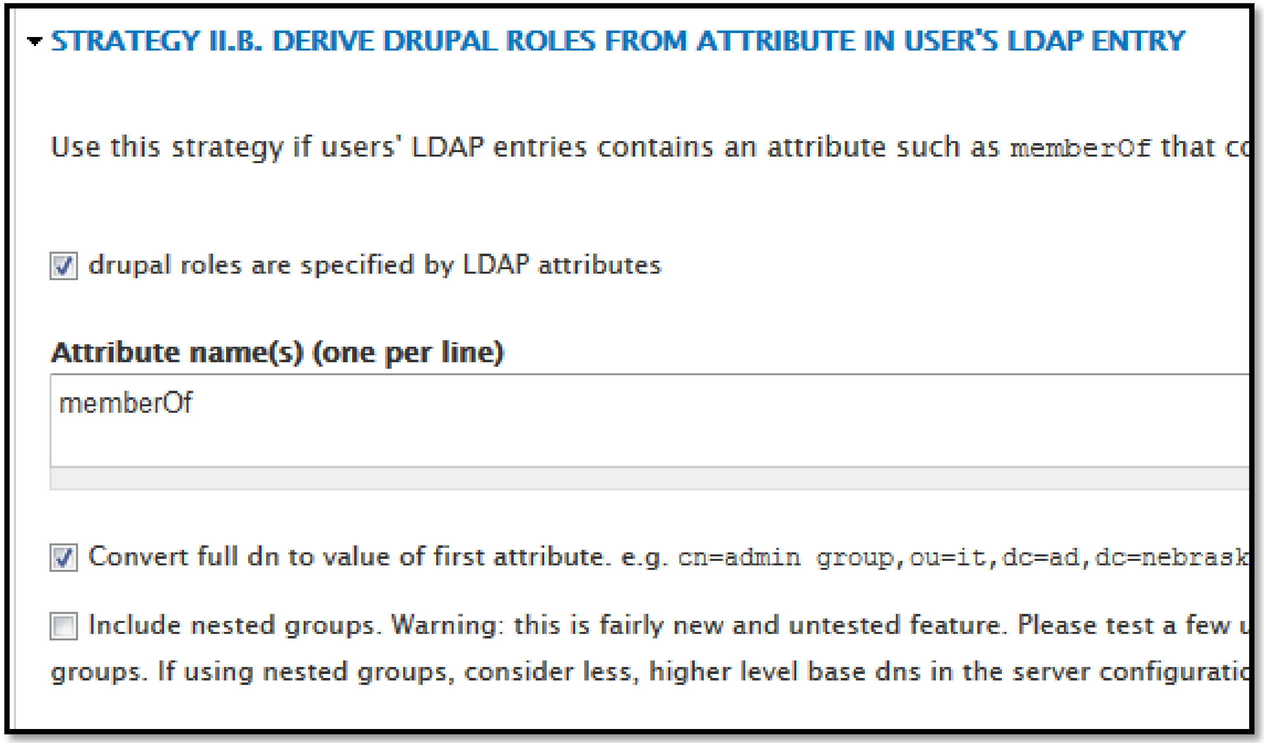
- Uses CornellAD

Cons

- Only applies application authorization
- Only works on Cornell campus

Authentication / Authorization with Drupal

Authorization: Drupal LDAP



▾ STRATEGY II.B. DERIVE DRUPAL ROLES FROM ATTRIBUTE IN USER'S LDAP ENTRY

Use this strategy if users' LDAP entries contains an attribute such as `memberOf` that co

☑ drupal roles are specified by LDAP attributes

Attribute name(s) (one per line)

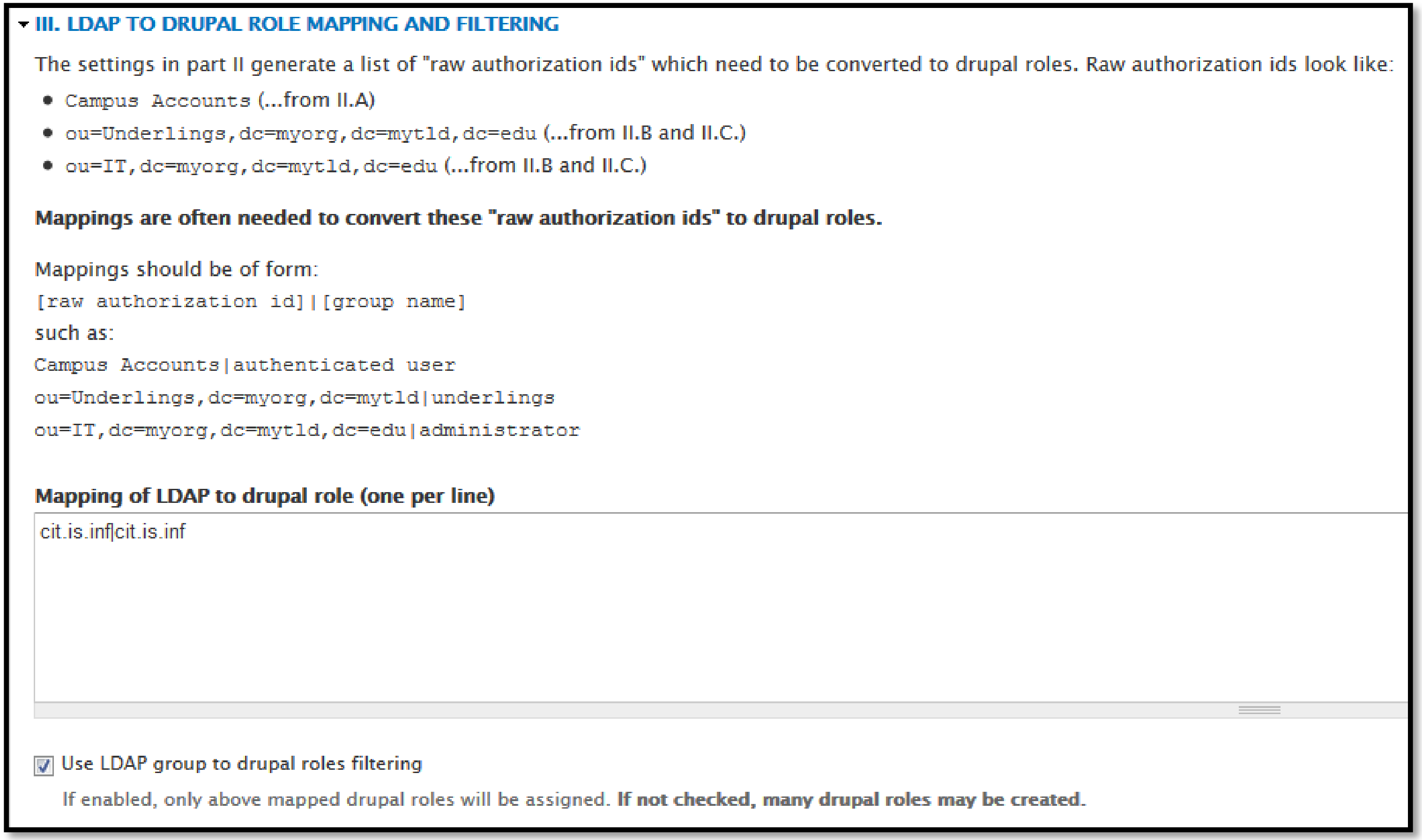
memberOf

☑ Convert full dn to value of first attribute. e.g. `cn=admin group,ou=it,dc=ad,dc=nebrask`

☐ Include nested groups. Warning: this is fairly new and untested feature. Please test a few u groups. If using nested groups, consider less, higher level base dns in the server configuratio

Authentication / Authorization with Drupal

Authorization: Drupal LDAP



▾ III. LDAP TO DRUPAL ROLE MAPPING AND FILTERING

The settings in part II generate a list of "raw authorization ids" which need to be converted to drupal roles. Raw authorization ids look like:

- `Campus Accounts` (...from II.A)
- `ou=Underlings,dc=myorg,dc=mytld,dc=edu` (...from II.B and II.C.)
- `ou=IT,dc=myorg,dc=mytld,dc=edu` (...from II.B and II.C.)

Mappings are often needed to convert these "raw authorization ids" to drupal roles.

Mappings should be of form:

```
[raw authorization id]|[group name]
```

such as:

```
Campus Accounts|authenticated user
ou=Underlings,dc=myorg,dc=mytld|underlings
ou=IT,dc=myorg,dc=mytld,dc=edu|administrator
```

Mapping of LDAP to drupal role (one per line)

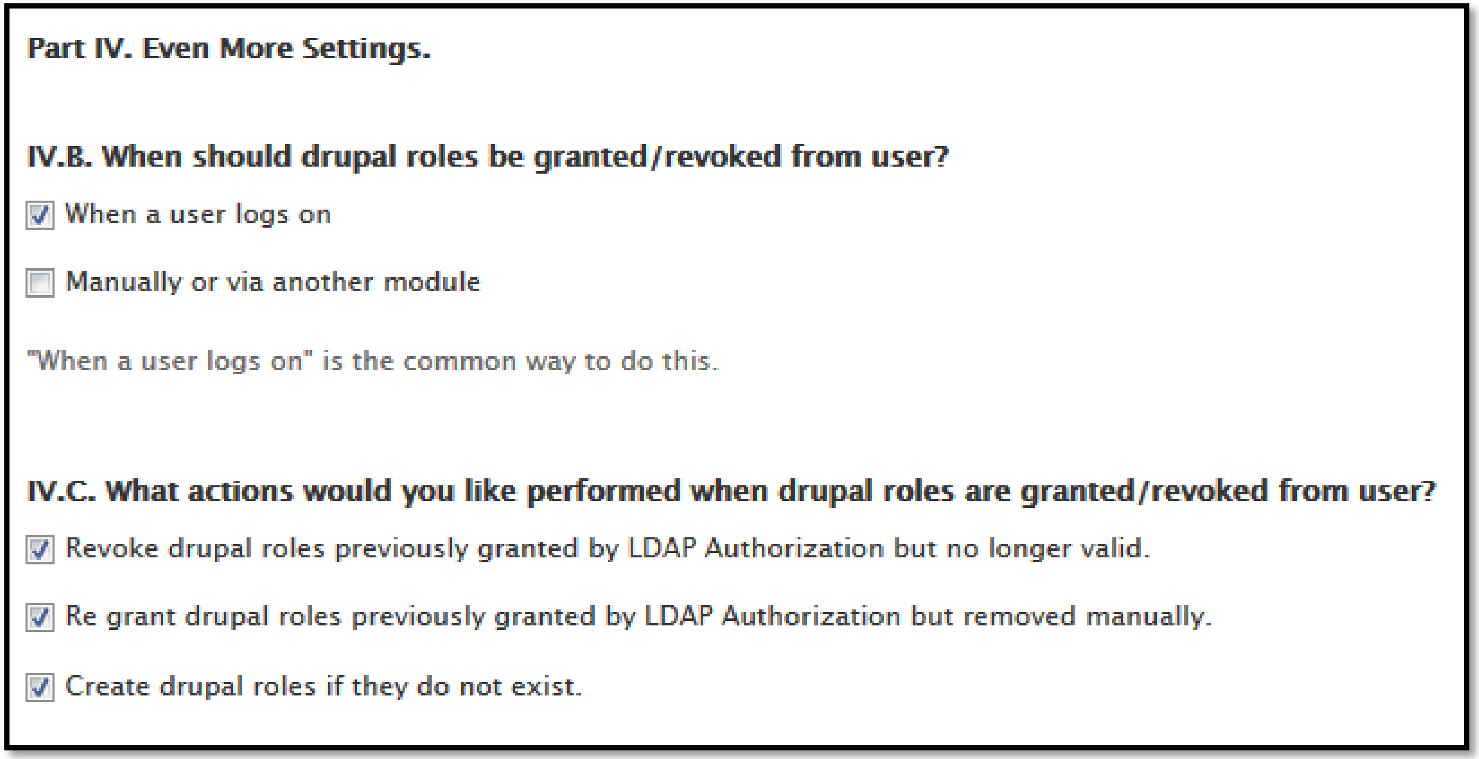
```
cit.is.inf|cit.is.inf
```

☑ Use LDAP group to drupal roles filtering

If enabled, only above mapped drupal roles will be assigned. **If not checked, many drupal roles may be created.**

Authentication / Authorization with Drupal

Authorization: Drupal LDAP



Part IV. Even More Settings.

IV.B. When should drupal roles be granted/revoked from user?

☑ When a user logs on

□ Manually or via another module

"When a user logs on" is the common way to do this.

IV.C. What actions would you like performed when drupal roles are granted/revoked from user?

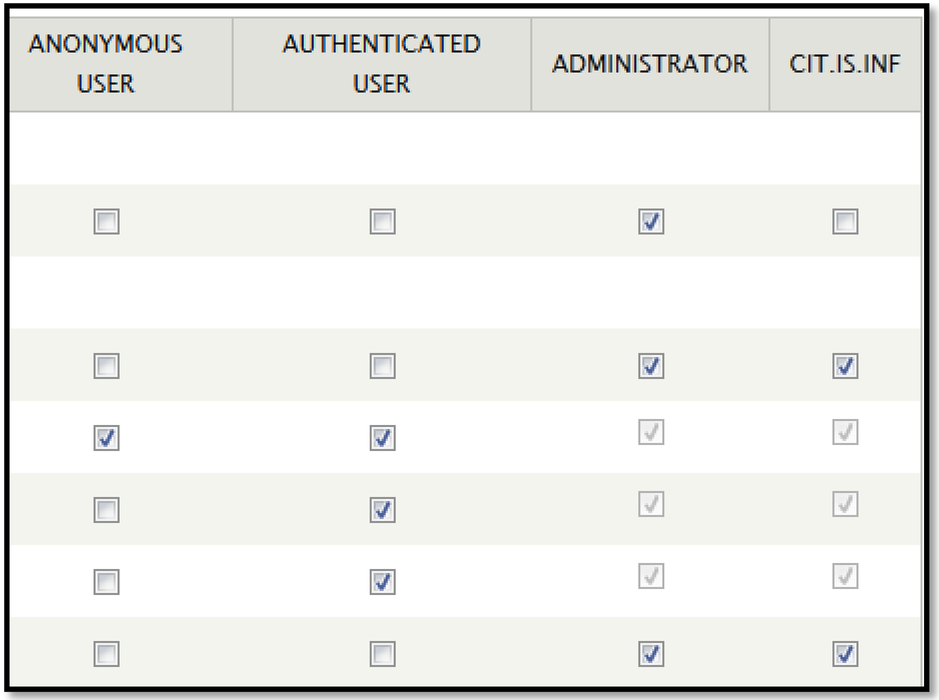
☑ Revoke drupal roles previously granted by LDAP Authorization but no longer valid.

☑ Re grant drupal roles previously granted by LDAP Authorization but removed manually.

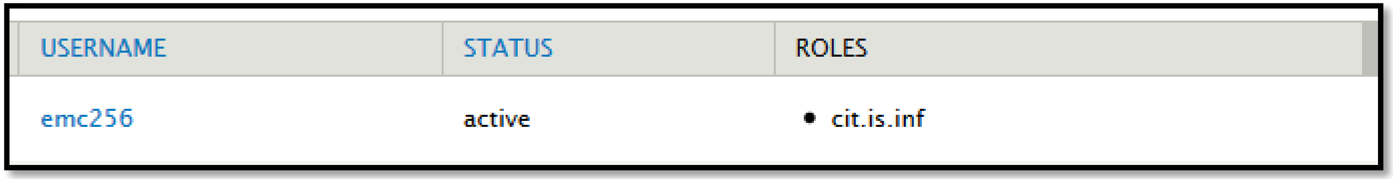
☑ Create drupal roles if they do not exist.

Authentication / Authorization with Drupal

Authorization: Drupal LDAP



ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	CIT.IS.INF
□	□	☑	□
□	□	☑	☑
☑	☑	☑	☑
□	☑	☑	☑
□	☑	☑	☑
□	□	☑	☑



USERNAME	STATUS	ROLES
emc256	active	• cit.is.inf

Authentication / Authorization with Drupal

Authorization: Other Options

- Manually map IDs to roles
- Custom code / mapping