

Grouper rules pattern - Assign attribute to folder

This is here for historical purposes.

If a stem is created somewhere and has etc or ref as an extension, add the do_not_provision_to attribute with values of "ad" and "ldap"

Note, make sure you are not using "smart quotes or dashes". You can disable these in a Mac in Preferences->keyboard-text.

Setup


First off you need GrouperSystem to be able to access the API in rules:

grouper.properties:

```
# configure some group
rules.accessToApiInEl.group = etc:hasAccessToApiInEl
```

put GrouperSysAdmin in that group:

[Home](#) > [Root](#) > [etc](#) > [hasAccessToApiInEl](#)

 **hasAccessToApiInEl**

[+ Add members](#)

[More actions ▾](#)


[More ▾](#)

[Members](#) [Privileges](#) [More ▾](#)

The following table lists all entities which are members of this group.

Filter for: [Apply filter](#) [Reset](#)

[Remove selected members](#)

| <input type="checkbox"/> Entity name ▾ | Membership | Choose action |
|--|------------|---------------------------|
| <input type="checkbox"/>  GrouperSysAdmin | Direct | Actions ▾ |

Show:

Showing 1-1 of 1 · [First](#) | [Prev](#) | [Next](#) | [Last](#)

Example1: UI example of a multi-assign single-valued attribute (this IS what PSPNG does)

Create a rule on some ancestor folder:

Home > Root > root > app

app

The app folder contains folders which contain resources for enterprise services. For example there might be a "vpn" or "wiki" folder in the "app" folder.

More ▾

Folder contents

Privileges

Edit folder

More actions ▾

Attribute Assignments

+ Assign attribute

The following table lists all attributes assigned to this folder

| Assignment type | Attribute name | Enabled? | Assignment values |
|------------------------|--------------------------|----------|---|
| Direct assignment | rule | enabled | |
| Metadata on assignment | ruleActAsSubjectId | enabled | GrouperSystem ▾ |
| Metadata on assignment | ruleActAsSubjectSourceId | enabled | g:isa ▾ |
| Metadata on assignment | ruleCheckStemScope | enabled | SUB ▾ |
| Metadata on assignment | ruleCheckType | enabled | stemCreate ▾ |
| Metadata on assignment | ruleIfConditionEl | enabled | `\${"etc".equals(stem.getExtension()) "ref".equals(stem.getExtension())}` ▾ |
| Metadata on assignment | ruleThenEl | enabled | `\${ { stem.getAttributeValueDelegate().assignValueString("etc:pspng:do_not_provision_to", "ldap"); stem.getAttributeDelegate().addAttributeByName("etc:pspng:do_not_provision_to").getAttributeAssign().getValueDelegate().assignValueString("ad"); } }` ▾ |
| Metadata on assignment | ruleValid | enabled | T ▾ |

Text: note: there are two sets of curlies because if you have multiple statements you need to put them in another set of curlies

```

rule
ruleActAsSubjectId      GrouperSystem
ruleActAsSubjectSourceId g:isa
ruleCheckStemScope      SUB
ruleCheckType           stemCreate
ruleIfConditionEl       `${"etc".equals(stem.getExtension()) || "ref".equals(stem.getExtension())}`
ruleThenEl              `${ { stem.getAttributeValueDelegate().assignValueString("etc:pspng:
do_not_provision_to", "ldap"); stem.getAttributeDelegate().addAttributeByName("etc:pspng:do_not_provision_to").
getAttributeAssign().getValueDelegate().assignValueString("ad"); } }`
ruleValid               T

```

If I create a stem in that parent folder with "etc" or a "ref" as an extension, it will get some attributes

Home > New folder

New folder

Create in this folder:

root:app:workflow

Enter a folder name or [search for a folder where you are allowed to create new folders](#).
Enter 'Root' for the top level folder

Folder name:

etc

Name is the label that identifies this folder, and might change.

Folder ID:

etc

☐ Edit the ID

ID is the unique identifier for this folder. It should be short and simple, and might have character restrictions. The ID should rarely change, if ever.

Description:

Description contains notes about the folder, which could include: what the folder represents, why it was created, etc.

Save

Cancel

Home > Root > root > app > workflow > etc

etc

Edit folder

More actions ▼

More ▼

Folder contentsPrivileges

Attribute Assignments

+ Assign attribute

The following table lists all attributes assigned to this folder

| Assignment type | Attribute name | Enabled? | Assignment values | Attribute definition | Choose action |
|-------------------|---------------------|----------|-------------------|-------------------------|---------------|
| Direct assignment | do_not_provision_to | enabled | ldap ▼ | do_not_provision_to_def | Actions ▼ |
| Direct assignment | do_not_provision_to | enabled | ad ▼ | do_not_provision_to_def | Actions ▼ |

Example2: UI example of a single-assign multi-valued attribute (this is NOT what PSPNG does)

Create a rule on some ancestor folder:

Home > Root > root > app

app

The app folder contains folders which contain resources for enterprise services. For example there might be a 'vpn' or 'wiki' folder in the "app" folder.

Edit folder

More actions ▾

More ▾

Folder contents

Privileges

Attribute Assignments

+ Assign attribute

The following table lists all attributes assigned to this folder

| Assignment type | Attribute name | Enabled? | Assignment values |
|------------------------|--------------------------|----------|---|
| Direct assignment | rule | enabled | |
| Metadata on assignment | ruleActAsSubjectId | enabled | GroupersSystem ▾ |
| Metadata on assignment | ruleActAsSubjectSourceId | enabled | g:isa ▾ |
| Metadata on assignment | ruleCheckStemScope | enabled | SUB ▾ |
| Metadata on assignment | ruleCheckType | enabled | stemCreate ▾ |
| Metadata on assignment | ruleIfConditionEl | enabled | `\${"etc".equals(stem.getExtension()) "ref".equals(stem.getExtension())}` ▾ |
| Metadata on assignment | ruleThenEl | enabled | `\${(stem.getAttributeValueDelegate().addValueString("etc:pspng:do_not_provision_to", "ldap"); stem.getAttributeValueDelegate().addValueString("etc:pspng:do_not_provision_to", "ad");)}` ▾ |
| Metadata on assignment | ruleValid | enabled | T ▾ |

Text: note: there are two sets of curlies because if you have multiple statements you need to put them in another set of curlies

```

rule
ruleActAsSubjectId      GrouperSystem
ruleActAsSubjectSourceId g:isa
ruleCheckStemScope      SUB
ruleCheckType           groupCreate
ruleIfConditionEl        ${groupName.contains(":etc:") || groupName.contains(":ref:")}
ruleThenEl              ${ { stem.getAttributeValueDelegate().addValueString("etc:pspng:do_not_provision_to",
"ldap"); stem.getAttributeValueDelegate().addValueString("etc:pspng:do_not_provision_to", "ad"); } }
ruleValid               T

```

If I create a folder in that parent folder or subfolder with an extension of "etc" or a "ref", it will get some attributes

New folder

Create in this folder:

Enter a folder name or [search for a folder where you are allowed to create new folders](#).

Enter 'Root' for the top level folder

Folder name:

Name is the label that identifies this folder, and might change.

Folder ID:

☐ Edit the ID

ID is the unique identifier for this folder. It should be short and simple, and might have character restrictions. The ID should rarely change, if ever.

Description:

Description contains notes about the folder, which could include: what the folder represents, why it was created, etc.



More ▾

Attribute Assignments

The following table lists all attributes assigned to this folder

| Assignment type | Attribute name | Enabled? | Assignment values | Attribute definition | Choose action |
|-------------------|---------------------|----------|-------------------|-------------------------|--|
| Direct assignment | do_not_provision_to | enabled | ldap ▾ ad ▾ | do_not_provision_to_def | <input type="button" value="Actions"/> |