Proposed Permission Set Guidelines

- Overview
- Naming Conventions
- Backend Modules
- Visible Permission Sets
- UI Modules
- Using *.all Permissions
- Other Considerations
- Appendix
  - Examples of Permission Sets with Misleading Names

Overview

One outcome of a recent investigation into misleading permission set configuration is the proposal of new guidelines for permission sets. This document serves as a place to capture that proposal and elicit feedback from others and facilitate discussion.

Naming Conventions

1. Name the permission sets appropriately, e.g. avoid inclusion of POST/PUT/DELETE permissions in "*.view" permission sets.
   - Clear:

```json
{
  "permissionName": "ui-erm-usage.view-create-edit",
  "displayName": "eUsage: Can view, create and edit usage data providers and COUNTER reports",
  "id": "ade748af-66b5-4584-a319-3cac20899241",
  "description": "Can view, create and edit usage data providers and COUNTER reports",
  "tags": [],
  "subPermissions": [
    "module.erm-usage.enabled",
    "usagedataproviders.collection.get",
    "usagedataproviders.item.get",
    "usagedataproviders.item.post",
    "usagedataproviders.item.put",
    "counterreports.collection.get",
    "counterreports.item.get",
    "counterreports.item.post",
    "counterreports.item.put",
    "aggregatorsettings.collection.get",
    "aggregatorsettings.item.get"
  ],
  "childOf": [],
  "grantedTo": [
    "fba0106d-e2ad-494e-8958-ce5b447ab2aa"
  ],
  "mutable": false,
  "visible": true,
  "dummy": false
}
```

- Misleading:
1. "permissionName": "ui-receiving.basic.view",
   "displayName": "Receiving: Basic view",
   "id": "5542bb26-4eff-4699-a7c5-6e6a049979d7",
   "tags": [],
   "subPermissions": [
     "module.receiving.enabled",
     "orders.item.get",
     "orders.pieces.item.post",
     "orders.pieces.item.put",
     "orders.po-lines.collection.get",
     "orders.titles.collection.get",
     "orders.titles.item.get",
     "ui-receiving.third-party-services"
   ],
   "childOf": [
     "ui-receiving.view"
   ],
   "grantedTo": [],
   "mutable": false,
   "visible": false,
   "dummy": false
}

Permission sets for modulePermissions should use the "modperms" prefix. These also should not be visible as they aren't intended to be assigned to users.

Example:

"permissionName": "modperms.circulation.loans.anonymize",
"permissionName": "modperms.circulation.override-renewal-by-barcode.post",
"permissionName": "modperms.circulation.renew-by-barcode.post",
"permissionName": "modperms.circulation.requests.item.move.post",
"permissionName": "modperms.circulation.renew-by-id.post",
"permissionName": "modperms.circulation.requests.item.post",
"permissionName": "modperms.circulation.override-check-out-by-barcode.post",
"permissionName": "modperms.circulation.requests.instances.item.post",
"permissionName": "modperms.circulation.check-out-by-barcode.post",
"permissionName": "modperms.orders.item.post",
"permissionName": "modperms.orders.item.put",

Backend Modules

- Avoid inclusion of other modules permissions in your permission sets. For example, mod-foo's permission set foo.all shouldn't include mod-bar's bar.item.get permission. Here it's the module that needs the permission, not the user.
  - Do: include the other module's permission(s) in your modulePermissions (or non-visible modulePermission sets - see above)
  - Don't: include other modules permissions in your visible permission sets that will be assigned to users.

Visible Permission Sets

- Be careful about making a permission set visible. Permissions should only have visible=true if/when they're intended to be granted to a user. Permission sets for modulePermissions should not be visible.

UI Modules

- Define separate permission sets for settings if other other module permissions are needed (e.g. configuration.entries.collection.get).
  - Example: "ui-users.settings.customfields.edit" probably needs configuration.entries.collection.get, but "ui-users.view" probably doesn't. If needed, additional permission sets should be created with appropriate names.
  - Example: Should we really be granting users, source-storage, circulation, configuration, etc. permissions from a ui-inventory permission set?
Using *.all Permissions

- Only include *.all permissions when absolutely sure it's necessary/appropriate. Instead use just the permissions actually needed.
  - Example of a permission set that probably abuses *.all permissions
Other Considerations

- Sensitive information should NEVER be stored in mod-configuration. SMTP credentials, RMAPI credentials, etc. should be stored behind their own endpoints which are protected with distinct permissions.
- We might want to take a look at ways to make the permissions of this module more granular, perhaps something could be done using desiredPermissions / X-Okapi-Permissions? Though I'm not sure this is really doable or even worth it. I think I'm in favor of just moving away from mod-configuration altogether (see below)
- We should consider moving away from using mod-configuration in general. The cross-app nature of this module makes it difficult to deal with other things like sample/reference data... what if you want some reference data loaded into mod-configuration, but not all of it?
  - See FOLIO-2583 - Getting issue details...
  - One use case where it probably makes sense to continue to have centralized configuration is for things that are truly shared across multiple apps, e.g. the tenant settings
- How do we handle "migration" of permission sets as teams clean things up?
  - One idea is to do something like add a field to the module descriptor that allows you to specify that this permission set "replaces" one named XYZ. This would result the swapping and cleanup of permissions.
  - See FOLIO-2607 - Getting issue details...

Appendix

Examples of Permission Sets with Misleading Names

```json
{
    "permissionName": "ui-checkin.all",
    "displayName": "Check in: All permissions",
    "id": "094310c8-cd71-4b76-a10d-2921ccd10654",
    "description": "Entire set of permissions needed to use Checkin",
    "tags": [],
    "subPermissions": [
        "circulation.all",
        "circulation-storage.all",
        "configuration.all",
        "users.collection.get",
        "usergroups.collection.get",
        "module.checkin.enabled",
        "inventory.items.collection.get",
        "inventory-storage.service-points.collection.get"
    ],
    "childOf": [],
    "grantedTo": [
        "fba0106d-e2ad-494e-8958-ce5b447ab2aa"
    ],
    "mutable": false,
    "visible": true,
    "dummy": false
}
```
"childOf": [
  "ui-receiving.view"
],
"grantedTo": [],
"mutable": false,
"visible": false,
"dummy": false
},
"permissionName": "ui-organizations.basic.view",
"displayName": "Organizations: Basic view",
"id": "c0f97cde-7fb5-46b1-814c-d78b58d62da0",
"tags": [],
"subPermissions": [
  "module.organizations.enabled",
  "organizations-storage.accounts.collection.get",
  "organizations-storage.accounts.item.get",
  "organizations-storage.addresses.collection.get",
  "organizations-storage.addresses.item.get",
  "organizations-storage.agreements.collection.get",
  "organizations-storage.agreements.item.get",
  "organizations-storage.alliances.collection.get",
  "organizations-storage.alliances.item.get",
  "organizations-storage.categories.collection.get",
  "organizations-storage.categories.item.get",
  "organizations-storage.contacts.all",
  "organizations-storage.emails.collection.get",
  "organizations-storage.emails.item.get",
  "organizations-storage.interaces.all",
  "organizations-storage.organizations.collection.get",
  "organizations-storage.organizations.item.get",
  "organizations-storage.phone-numbers.collection.get",
  "organizations-storage.phone-numbers.item.get",
  "organizations-storage.urls.collection.get",
  "organizations-storage.urls.item.get",
  "ui-organizations.third-party-services"
],
"childOf": [
  "ui-organizations.view",
  "ui-organizations.creds.view"
],
"grantedTo": [],
"mutable": false,
"visible": false,
"dummy": false
},
"permissionName": "ui-licenses.licenses.view",
"displayName": "Licenses: Search & view licenses",
"id": "9f9bad1c-dfbd-4c36-98f6-7ef9c8c722d6",
"tags": [],
"subPermissions": [
  "module.licenses.enabled",
  "licenses.item.post",
  "licenses.licenses.view",
  "licenses.files.view",
  "licenses.contacts.view",
  "licenses.custprops.view",
  "licenses.orgs.view"
],
"childOf": [
  "ui-licenses.licenses.edit",
  "ui-licenses.licenses.delete"
],
"grantedTo": [],
"mutable": false,
"visible": true,
"dummy": false
},
"permissionName": "ui-notes.item.view",
"displayName": "Notes: Can view a note",
"id": "ccb872ae-eb1b-421d-9b36-4f818e96bde7",
"tags": []}
"subPermissions": [
  "note.types.collection.get",
  "notes.item.get",
  "notes.collection.get",
  "notes.collection.get.by.status",
  "notes.domain.all",
  "module.notes.enabled"
],
"childOf": [
  "ui-notes.item.create",
  "ui-notes.item.edit",
  "ui-notes.item.delete",
  "ui-notes.item.assign-unassign"
],
"grantedTo": [],
"mutable": false,
"visible": true,
"dummy": false
],
{
  "permissionName": "ui-receiving.view",
  "displayName": "Receiving: View",
  "id": "33096278-520c-4268-8f2f-a5075e8e7171",
  "tags": [],
  "subPermissions": [
    "orders.check-in.collection.post",
    "orders.receiving.collection.post",
    "settings.receiving.enabled",
    "ui-receiving.basic.view"
  ],
  "childOf": [
    "ui-receiving.edit"
  ],
  "grantedTo": [],
  "mutable": false,
  "visible": true,
  "dummy": false
}