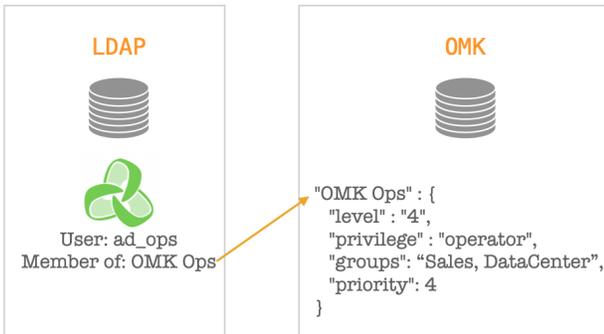


User Authorisation with Active Directory and LDAP

- Introduction
- Prerequisites
- Caveats
- Configuration
 - The mapping file
- Integrating with MS-LDAP

Introduction

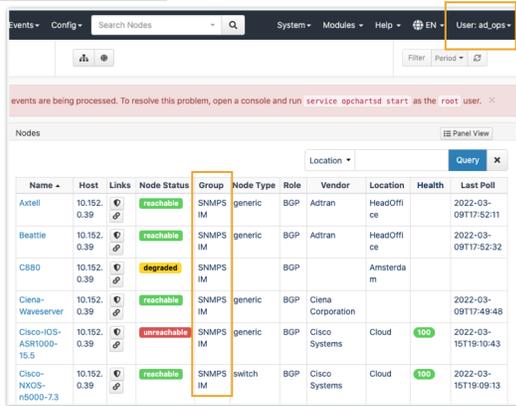
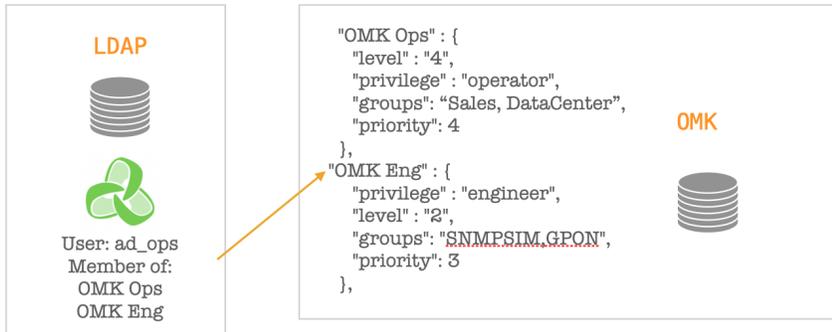
Authorisation with LDAP allow users to get privileges and groups assigned based on a LDAP group.



The screenshot shows a network management interface with a table of nodes. The 'Group' column is highlighted with an orange box. The table contains the following data:

Name	Host	Links	Node Status	Group	Node Type	Role	Vendor	Location	Health	Last Poll
meatball09	203.206.187.243		unreachable	DataCenter	generic	core		Cloud		
primary_imp_or_test2	127.0.0.1		reachable	Sales	server	distribution	net-snmp			2022-03-11T17:25:24
primary_imp_or_test3	127.0.0.1		reachable	DataCenter	server	access	net-snmp			2022-03-09T17:51:30
primary_imp_or_test4	127.0.0.1		reachable	DataCenter	server	access	net-snmp			2022-03-09T17:51:20
rt1	10.248.0.1		reachable	DataCenter	router	core	Cisco Systems	Cloud		2022-03-11T17:00:23
vyos-rt1	10.248.255.11		unreachable	Sales	generic	core		HeadOffice		

If a user belongs to more than one group, the privilege will be selected based on the priority (1 is higher priority than 10):



Prerequisites

- LDAP authentication must be set. See [Configuring NMIS to use Active Directory Authentication \(ms-ldap or ms-ldaps\)](#) and [OMK Authentication Methods](#) for further details.

Caveats

- Before opEvents 4.3.2 using this feature disables the use of the NMIS User Table completely.
- From opEvents 4.3.2, opCharts 4.5.5, this feature has been enhanced to allow local user privileges when auth_ldap_privs is enabled.
 - If local user exists, use privilege and groups for that local user from Users.nmis
 - If local user does **not** exist **and** we're using auth_ldap_privs, retrieve LDAP groups, match them to privilege and groups in AuthLdapPrivs.json
 - If we do **not** have a local user **and** we are **not** using LDAP **and** we have auth_default_privilege set, use that from opCommon.json. If we have auth_default_groups set, use that.

auth_default_privilege and auth_default_groups

When accessing NMIS, you have a choice on how to handle authenticated users who do not have authorisations defined, you can reject them, or you can allow them default access. This is so that you do not have to define every user in the system if the authentication system is providing a reduced list of users, to have the users become an operator or guest by default and be able to see all groups of devices, the following would apply: 'auth_default_privilege' => 'guest', 'auth_default_groups' => 'all'. To prevent default authorisation, simply define them as blank, which is the default in the NMIS Install configuration.

Configuration

Configuration items in opCommon.json

Item	Example Value	Description	Default
auth_ldap_privs	0/1	Set to 1 to enable the feature	0
auth_ldap_server	server.domain.com:389	The LDAP server	No defaults. Entry must be created.
auth_ldap_acc	administrator@domain.local	The LDAP account to be able to search	No defaults. Entry must be created.
auth_ldap_psw	Password	The password for being able to search	No defaults. Entry must be created.
auth_ldap_context	CN=Users,DC=opmantek,DC=local	The base search	No defaults. Entry must be created.

auth_ldap_group	memberOf	The attribute to lookup the group values. Must follow: CN=OMK Ops,CN=Users,DC=opmantek,DC=local	memberOf
-----------------	----------	--	----------

The mapping file

The mapping file by default, is named AuthLdapPrivs.json and it should be placed in <omk_dir>/conf.

It should contain a list of groups containing:

- privilege
- level
- groups
- priority

As an example:

```
{
  "OMK Admin" : {
    "privilege" : "administrator",
    "level" : "0",
    "groups": "all",
    "priority": 1
  },
  "OMK Eng" : {
    "privilege" : "engineer",
    "level" : "2",
    "groups": "SNMPSIM,GPON",
    "priority": 3
  }
}
```

You can find an example in <omk_dir>/install.

It is possible to change the default location/name in the configuration file opCommon.json:

auth_ldap_privs_file

Integrating with MS-LDAP

You need to use both auth_ms_ldap and auth_ldap attributes for this to be a successful integration, this is so we can query both our user and group base and then apply the correct RBAC roles.

For this to be successful with ms-ldap authentication, the following example below will guide you through the process. Note that in this example the LDAP base and context search has been set for the whole domain, *you can tune as you need to be more tightly integrated.*

/usr/local/omk/conf/opCommon.json

```
...
  "authentication" : {
    ...
    "auth_method_1" : "ms-ldap",
# First let's define the ms-ldap specific requirements
    "auth_ms_ldap_server" : "IP_ADDRESS_OF_YOUR_MS_LDAP_SERVER", #eg. 192.168.1.22
    "auth_ms_ldap_dn_acc" : "svc_omk_admin", # you should only need to use the username here, but if this is
not successful, you can use username@domain as well.
    "auth_ms_ldap_dn_psw" : "password_of_the_dn_acc_above",
    "auth_ms_ldap_base" : "dc=contoso,dc=local",
    "auth_ms_ldap_attr" : "sAMAccountName",
# Now we add in the ldap specific requirements, including enabling auth_ldap_privs
    "auth_ldap_privs" : 1,
    "auth_ldap_server" : "the_fqdn_of_your_ad_server:389", # you could also use an IP address here, but you
need to ensure that the LDAP/LDAPS port is added in the value, eg. 192.168.1.22:389
    "auth_ldap_acc" : "svc_omk_admin@contoso.local",
    "auth_ldap_psw" : "password_of_the_auth_ldap_acc_above",
    "auth_ldap_context" : "dc=contoso,dc=local",
    "auth_ldap_group" : "memberOf",
    ...
  },
...

```

Once saved, you will then need to restart the omkd daemon for this to take affect.

If your organisation uses ms-ldaps authentication, the process is very similar. You will need to replace the following:

```
"auth_ms_ldap_server" to "auth_ms_ldaps_server"
"auth_ldap_server" to "auth_ldaps_server"
```

Examples of integrating ms-ldap, ms-ldaps, ldap, ldaps and other authentication methods can be found here: [OMK Authentication Methods](#)