Invoker interface for Active Directory

Any agent, trigger or mapping can use the invoker method for the ActiveDirectory agent. The invoker method is available in the dispatcherService class.

The invoker method is not specific of the Active Directory agent. Many other connectors support this method. The expected arguments are:

- Action
- Object name
- Parameters

Here you have an example of a post-update trigger to create the home server for a user:

```
map = new HashMap();
String server = "//"+source{"homeServer"}+"/"+source{"accountName"};
// Create folder
f = dispatcherService.invoke("smb:mkdir", server, map);

// Add administrator ACL
map.put("user", "soffid_admin");
map.put("permission", "GENERIC_ALL");
map.put("flags", "CONTAINER_INHERIT_ACE OBJECT_INHERIT_ACE");
f = dispatcherService.invoke("smb:addacl", path, map);

// Add user ACL
map.put("user", source{"accountName"});
f = dispatcherService.invoke("smb:addacl", path, map);
// Change folder ownership using a domain admin account
map.put("_auth_user", "user1");
map.put("_auth_domain", "domain1");
map.put("_auth_password", "SuperSecret");
f = dispatcherService.invoke("smb:setOwner", path, map);
```

The example above uses the smb:mkdir action to create the folder, the smb:addacl to add a new access control list entry. Other commands allow the query and modification of Active Directory

objects like users and groups.

The list of allowed commands are:

Command	Object name	Parameters	Comments
insert	Object distinguished name	Object attributes	Creates a new active directory object
update	Object distinguished name	Object attributes	Modifies an existing active directory object. Only the attributes present in the map will be updated
delete	Object distinguished name	-	Removes an existing active directory object.
select	Base distinguished name	Object criteria attribute	Search for any object with the values specified in the parameters map, starting in the specified base DN. The return value is a list of maps. Each element in the list is an Active Directory object
get	Object distinguished name	-	Returns the object with the specified object DN. The return value is a list containing one or no maps. The map, if exists, contain the object attributes
smb:mkdir	Shared file	Optionally: • _auth_user • _auth_password • _auth_domain	Creates the shared folder. The shared folder name should follow the syntax //server/sharedFolder/Path or \\server\\sharedFolder\Path It is recommended to use the first syntax because the second one requires the script to escape any backslash character, leading to a harder to read script
smb:exist	Shared file	Optionally: • _auth_user • _auth_password • _auth_domain	Returns a list with a single map. The map has the attribute exist with a boolean value indicating whether the file exists or not

Command	Object name	Parameters	Comments
smb:rmdir	Shared file	Optionally: • _auth_user • _auth_password • _auth_domain	Removes the full directory and any file or directory within
smb:rm	Shared file	Optionally: • _auth_user • _auth_password • _auth_domain	Removes the file or directory. The command will fail if the directory is not empty.

Command	Object name	Parameters	Comments
smb:getacl	Shared file	Optionally: • _auth_user • _auth_password • _auth_domain	Returns a list of maps representing each access control list entry for that file or folder. Each map has three values: • user: The user or group name. When the user or group is unknown, the user or group SID is used. • permission: A text string with the permissions granted with that ACE. The string contains one or more of these values concatenated: ◦ FILE_READ_DATA ◦ FILE_WRITE_DATA ◦ FILE_APPEND_DATA ◦ FILE_EXECUTE ◦ FILE_LIST_DIRECTORY ◦ FILE_ADD_FILE ◦ FILE_ADD_SUBDIRECTORY ◦ FILE_TRAVERSE ◦ FILE_DELETE_CHILD ◦ FILE_READ_ATTRIBUTES ◦ FILE_WRITE_ATTRIBUTES ◦ FILE_READ_EA ◦ FILE_WRITE_EA ◦ DELETE ◦ READ_CONTROL ◦ WRITE_DAC ◦ WRITE_O

Command	Object name	Parameters	Comments
smb:addacl	Shared file	Map with these three values: • user • permission • flags And optionally, these ones: • _auth_user • _auth_password • _auth_domain	Adds an access control list with the specified permission and flags
smb:removeacl	Shared file	Map with these three values: • user • permission • flags And optionally, these ones: • _auth_user • _auth_password • _auth_domain	Remove the access control list entry that matches the map. If the permission or flag is missing, the connector will remove any access control list entry for the specified user
smb:setowner	Shared file	Map with the value: • user And optionally, these ones: • _auth_user • _auth_password • _auth_domain	Sets the directory owner to the one specified in the map

By default, actions are performed by the account used to configure the AD connector, but sometimes, another account must be used, mainly when dealing with NAS servers. In order to user custom credentials SMB commands accept three special parameters: _auth_user, _auth_password and _auth_domain. If this parameters are null, the agent user and password is used.

Revision #7
Created 22 April 2021 13:32:10 by pgarcia@soffid.com
Updated 27 June 2022 14:55:35 by pgarcia@soffid.com