Synchronizing Trustelem Groups with the Enterprise Vault CLI

□□ Goal

Automatically define access rights to **shared vaults** and **collections** in Enterprise Vault based on membership in **Trustelem groups**, using a custom JSON attribute called `vaultSync`.

□□ Overall Workflow

The script reads all Trustelem groups that include the `vaultSync` attribute.

1. For each Trustelem group with the attribute:
   - A **shared vault** is created or updated.
   - **Collections** can be created or removed according to the configuration.
   - **User permissions** are applied.
2. If a user also has a `vaultSync` attribute, it **overrides** the group configuration.

□□ Trustelem Configuration

Step 1 – Add custom attributes

□`vaultSync` JSON attribute on a **Trustelem Group** (mandatory)

Example to adapt:

```
[
  {
    "sharedVault": "SharedVault1",
```

```
    "mode": "exact",
    "collections": ["Collec1", "Collec2"],
    "createCollections": true,
    "deleteCollections": false,
    "role": "User",
    "permission": "view"
  },
  {
    "sharedVault": "SharedVault2",
    "mode": "exact",
    "collections": ["Colletion1", "Collection2"],
    "createCollections": true,
    "deleteCollections": false,
    "role": "User",
    "permission": "edit"
  }
]
```

Field	Type	Description
`sharedVault`	`string`	Name of the shared vault to sync
`mode`	`string`	"exact" or "pattern". Not mandatory, default to "exact" If "exact" is used, "collections" list must be exact names of synchronized collections if "pattern" is used, "collections" list is patterns for collections names.
`collections`	`string[]`	List of collections name or patterns
`createCollections`	`boolean`	Auto-create missing collections (can only be used with exact mode)
`deleteCollections`	`boolean`	Auto-remove collections not listed (can only be used with exact mode)
`role`	`string`	User role in the vault: `"Owner"`, `"Admin"`, or `"User"`
`permission`	`string`	Permission in the collection: `"view"`, `"viewExceptPass"`, `"edit"`, `"editExceptPass"`, `"manage"`

`createCollections` and `deleteCollections` are automatically considered to false with mode "pattern".

You can synchronize a group in one or more shared vaults.

You can synchronize one or more Trustelem groups in the same Shared Vault. By this way, you can affect different roles and rights to the differents groups of users.

[`vaultSync` JSON attribute on a **Trustelem User** (optional)

Overrides group settings when applied.
Example to adapt (the value needs to be an array):

```
[
  {
    "sharedVault": "SharedVault1",
    "mode": "pattern",
    "collections": ["collection1", "collection1/**"],
    "role": "Admin",
    "permission": "edit"
  }
]
```

> **How to use patterns ?**

Patterns uses **glob matching** syntax to consider collections names as filesystem path. It allow you to use wildcard and more complex syntax.

Glob matching - Using wildcards (`*` to match at one level of directories and `?` to replace one character), globstars (`**`) to include nested directories.

You can use POSIX character classes [:alpha:] or [:digit:], regex classes [1-5], regex logical (abc|xyz) and brace expansion. If you wont to have an exhausted documentation, you can refer to the library micromatch.

examples :

With collections tree as follow

```
collection1/
    collection11
    collection12/
        collection121
        collection122
collection2/
    collection21
```

Pattern	Matching results
`collection1/**`	`collection1`, `collection11`, `collection12`, `collection121`, `collection122` collection1 and all childs of collection1 with wildcard "**", parent is included

`*`	`collection1`, `collection2` all collection of first level
`collection?/*`	`collection11`, `collection12`, `collection21` all collection of second level with parent 'collection' and an other character. with wildcard "*", parent is not included
`*/collection[0-9]{1,2}`	all second level collections with a name 'collection' followed by one or two digits.

Step 2 – Generate a Trustelem API Key

1. In the Trustelem admin console, go to **Rest API > New API key**
   ⚠if you can't see the **Rest API** tab, you have to create a ticket to require the feature.
2. Click the **edit (pencil)** icon
3. Check:
   - `users_read`
   - `groups_read`
4. Save.
5. Note the following values:
   - **KEY ID**
   - **Bearer Token**

□□ CLI Script Configuration

1/ Download Enterprise Vault CLI.

2/ Create the `synchro` script

Example with the Linux CLI:

```
#!/bin/bash
# --- Configuration ---
vault_id="to_replace"
vault_secret="to_replace"
vault_password="to_replace"
vault_url="to_replace"
trustelem_key_id="to_replace"
trustelem_bearer="to_replace"
trustelem_url="to_replace"

# --- Function: Connect and unlock the Vault CLI ---
```

```
connect_to_vault() {
    ./wv logout > /dev/null 2>&1
    ./wv config server "$vault_url" > /dev/null 2>&1
    export WV_CLIENTID=$vault_id
    export WV_CLIENTSECRET=$vault_secret
    export WV_TRUSTELEMKEYID=$trustelem_key_id
    export WV_TRUSTELEMBEARER=$trustelem_bearer
    export WV_TRUSTELEMURL=$trustelem_url
    ./wv login --apikey > /dev/null 2>&1
    export WV_SESSION=$(./wv unlock "$vault_password" --raw)
}

# --- Main Execution ---
connect_to_vault
./wv syncgroups
```

PowerShell example with the Windows CLI:

```
# --- Configuration ---
$vault_id = "to_replace"
$vault_secret = "to_replace"
$vault_password = "to_replace"
$vault_url = "to_replace"
$trustelem_key_id = "to_replace"
$trustelem_bearer = "to_replace"
$trustelem_url = "to_replace"

# --- Function: Connect and unlock the Vault CLI ---
function Connect-ToVault {
    ./wv.exe logout | Out-Null
    ./wv.exe config server $vault_url | Out-Null
    $env:WV_CLIENTID = $vault_id
    $env:WV_CLIENTSECRET = $vault_secret
    $env:WV_TRUSTELEMKEYID = $trustelem_key_id
    $env:WV_TRUSTELEMBEARER = $trustelem_bearer
    $env:WV_TRUSTELEMURL = $trustelem_url
    ./wv.exe login --apikey | Out-Null
    $session = ./wv.exe unlock $vault_password --raw
    $env:WV_SESSION = $session.Trim()
}
```

```
# --- Main Execution ---
Connect-ToVault
./wv.exe syncgroups
```

Values to replace:

Variable	Description
`vault_id`	API service account ID (web client > Account parameters > Security > API Keys)
`vault_secret`	API service account secret (web client > Account parameters > Security > API Keys)
`vault_password`	Master password for the service account
`vault_url`	Vault instance URL (e.g., `https://vault-yourdomain.trustelem.com`)
`trustelem_key_id`	KEY ID from Step 2
`trustelem_bearer`	Bearer Token from Step 2
`trustelem_url`	Trustelem admin URL (e.g., `https://admin-yourdomain.trustelem.com`)

3/ Start the script.

⚙ Detailed Synchronization Behavior

1. Shared Vault Management

- **If the shared vault doesn't exist**:
  - It is **automatically created** and the service account becomes the `Owner`.
- **If it already exists**:
  - The script checks if the service account is the `Owner`.
  - If not, an **error is returned**.

⚠ **Important**: if the service account is not a shared vault member, it has no way to know if the shared vault already exists or not. In this case, the script will assume the shared vault doesn't exist and a new shared vault with the same name will be created.

2. Collection Management

- **With `createCollections=true`**:
  - Collections missing in the vault but listed in the JSON will be created.
  - The service account is given `manage` rights on these.
- **With `deleteCollections=true`**:
  - Collections present in the vault but not in the list will be deleted.
  - The service account must have `manage` rights to delete them.

⚠ **Important**: do not set both `createCollections=true` and `deleteCollections=true` at the same time.

Note on User Attributes: User-level `vaultSync` attributes **cannot** create or delete collections—only assign permissions.

3. User Membership & Permissions

- **If a user is not in Vault**:
  - An **error is returned**.
- **Adding users to collections**:
  - If the user is not already in a listed collection, they are added with the `role` and `permission` defined.
  - If the user has their own `vaultSync` config, it **overrides** the group config.
  - The service account must have `"manage"` rights.
- **Updating permissions**:
  - If the user has different permissions in a collection, they are updated accordingly.
  - Requires `"manage"` permission.
- **Removing from collections**:
  - If a user is in a collection not listed, they are removed.
  - Requires `"manage"` permission.

Special Cases

Scenario	Behavior
Trustelem group renamed in AD	Synchronization continues (attribute remains present)
Trustelem group deleted	Synchronization stops, but the vault remains unchanged
♻ Group deleted and recreated	Vault is no longer linked; re-adding the attribute resumes sync

---

Permissions Summary

Action	Required Right (Service Account)
Create shared vault	None (becomes `Owner`)
Modify existing vault	Must be `Owner`
Create/delete collection	`manage`
Add/remove users from collections	`manage`
Update user permissions	`manage`

Revision #17
Created 13 May 2025 13:35:41 by WALLIX Admin
Updated 1 December 2025 13:53:34 by WALLIX Admin