CloudEndure Migraiton Factory

ENHANCEMENTS

Table of Contents

Overview

Enhance existing CEMF migration script to minimize user input of CloudEndure Migration Factory account, respective server accounts on both Linux and Windows servers, and CloudEndure installation key. Manually inputting these accounts takes time and prone to user error.

This enhancements will minimize manual input of CEMF account, server accounts, and CloudEndure installation key to improve efficiency of the users.

Enhancements

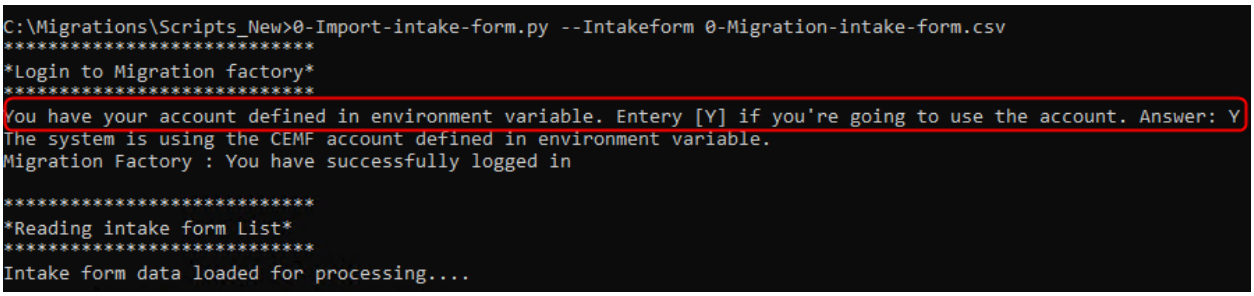
CloudEndure Migration Factory (CEMF) Account Entry

All the scripts that requires CEMF account entry are updated by allowing user to set CEMF username and password in environment variable. If the script detects that there is CEMF account defined in environment variable, it will prompt the user whether to use the account or enter the account manually.

If the environment variables are not set it will behave to its default behavior which is to ask the account.

You must set this environment variable before you execute the script. You don't need to set it again when you execute next script as long as you do not close and re-open the console. See How to Verify if the Environment Variable Exists.

In the image blow, it shows the question being ask by the script once CEMF account information is detected. If the user replied *Y*, it will proceed using the predefined account. It will also notify the user that the predefined account is used.



Limitation

The environment variable only exist within the session of the command console. If you open a new command console or you close and re-open the console, you need to re-enter the environment variables again.

How to Set the Environment Variables

Use *SET* command of the windows command console to set the environment variable. For CEMF account we have to set the following environment variable.

- CEMF_USERNAME. Your CloudEndure Migration Factory username.
- CEMF_PASSWORD. Your CloudEndure Migration Factory password.

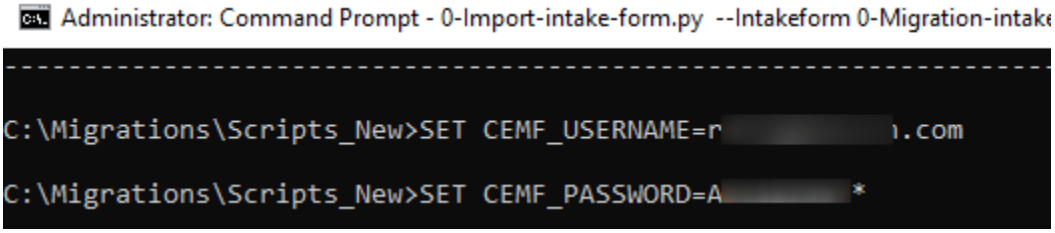
Use *SET* command to set these environment variables with your CEMF account follow this script:

```
SET CEMF_USERNAME=<your CEMF username>
```

```
SET CEMF_PASSWORD=<your CEMF password>
```

Example:

```
SET CEMF_USERNAME=rod@mydomain.com
SET CEMF_PASSWORD=Abc123def*
```



How to Verify if the Environment Variable Exists

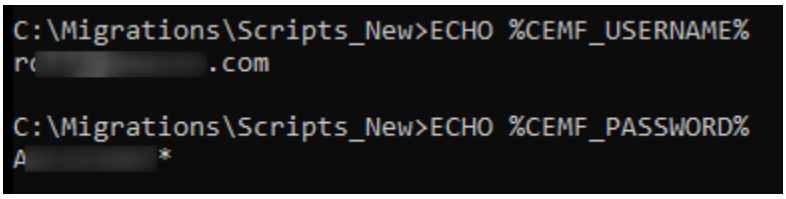
To verify if you have existing environment variable set, use *ECHO* command.

```
ECHO %<environment variable>%
```

Example:

To display the CEMF username or password, use this script.

```
ECHO %CEMF_USERNAME%
ECHO %CEMF_PASSWORD%
```



Linux and Windows Account Entry

Other improvement brought by this enhancement, it is more interactive with the user by providing the following options:

- Enter the account manually
- Use the previously entered account
- Use the previously entered account to all remaining servers.

Linux and windows servers account are stored separately.

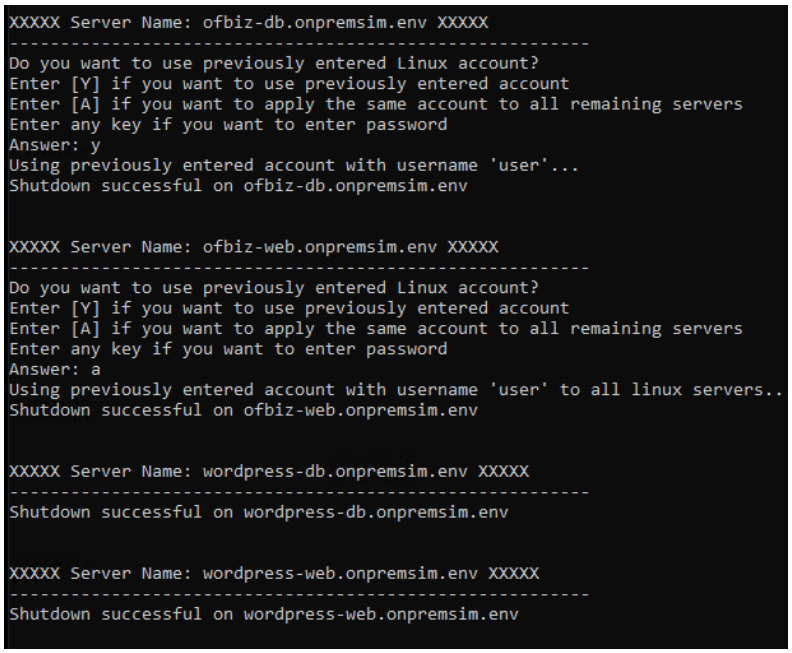
How It Works

The script store the account information automatically every time the user enter the account in the console. The data is stored in *cemf_settings.json* file. On the succeeding request the script will ask the user to:

- **Enter the account manually (Default)**. If the user chooses this option, it will allow the user to enter the account. This account will be saved that way it can be used in the next request.
- **Use the previously entered account (Choose: Y)**. If the user chooses this option, it will allow the user to use the previously entered account to save time and minimize error in manually inputting the account. The script will ask again before processing the next server.
- **Use the previously entered account to all remaining servers (Choose: A)**. If the user chooses this option, it will allow the user to use the same account for all succeeding servers to be process.

If the script did not detect account information or it's your first time to use the script, it will use the default behavior.

In the screenshots, you can see how the system behave when **Y** (Using Previously entered account) is selected and **A** (Apply the same account to all remaining servers).



Limitations

Currently the data is stored in *cemf_settings.json* file are **not** encrypted.

Sample Data:

```
{
      "linuxUsername": "user",
      "linuxPassword": "password",
      "linuxHas_key": "N",
      "windowsUsername": "user",
      "windowsPassword": "password",
      "windowsHas_key": "N"
      ...
      ...
}
```

Make sure to manually delete this file after using the scripts. This feature enhancements is added in the Recommended Future Enhancements.

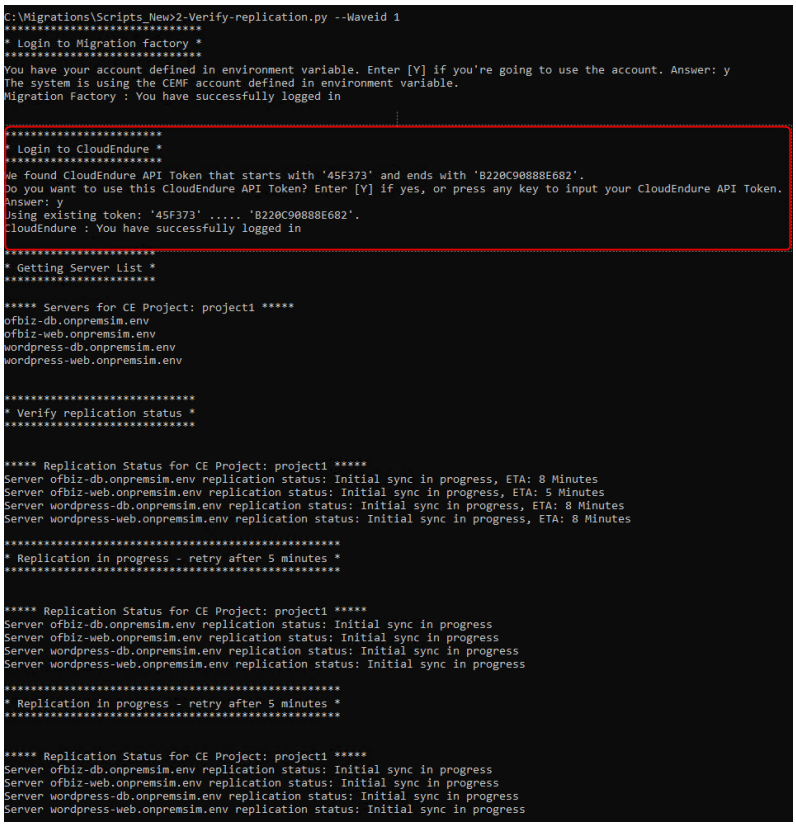
CloudEndure API Token Enahcement

CloudEndure API Token is a key that is use to allows Migration Factory to communicate with CloudEndure through API. There are 5 CEMF scripts that requires you to manually enter the 64 digit CloudEndure API token that is generated by *0-CEProjects.py* script. User has to copy it manually from the console that way the user can use it in other script that may require it.

How It Works

When CEMF creates CloudEndure project (*0-CEProjects.py*), it creates CloudEndure project and returns the CloudEndure API token. This token is automatically stored that will be consumed by 5 CEMF scripts that is dependent to CloudEndure API Token.

When CloudEndure API token is detected, it will prompt the user whether to use it or to manually input the CloudEndure API token. If the user choose to use the previously saved token, it will automatically use the token to execute the script else the user is required to manually input the 64 character token.



Limitations

Currently the CloudEndure API token is stored in *cemf_settings.json* file and not encrypted. This file is auto-generated by the script once account is inputted.

Sample Data:

```
{
      "CEApiKey": "XYSDJD6. . . UJDH",
      ...
      ...
}
```

Latest updates on Intake From CSV (0-Migration-intake-form.csv)

Latest intake form CSV has *aws_account* column added. If you are still using the old version of intake form, you need to update the *0-Import-intake-form.py* script with the one in ***workshop*** folder included in this package

Using SSH Private Key in connecting to Linux Machine

If you want to use SSH private key in connecting to Linux machine you can follow this steps.

All you have to do:

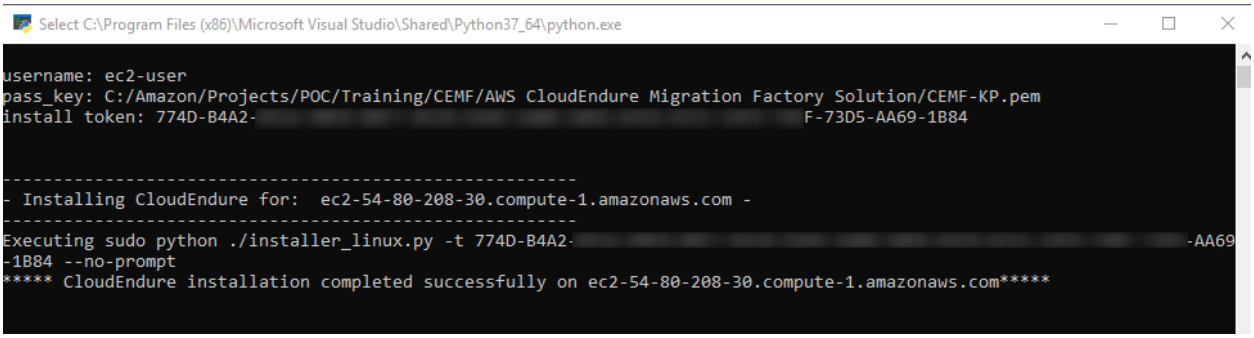
- Answer ***Y*** when prompted "*If you use a private key to login, press [Y] or if use password press [N]*".

```
*******************************************
* Enter Linux Sudo username and password *
*******************************************
Linux Username: user
If you use a private key to login, press [Y] or if use password press [N]: n
Linux Password:
Re-enter Password:
```

- In the *pass_key* field instead of entering the password, Enter the *path* where the SSH private key file is located in your local machine.

The SSH private key (*.pem) is downloaded when you create EC2 key pair.

Testing Output:



Recommended Future Enhancements

Safely Store Account Information

Use AWS Secret Manager to store the data securely to avoid security issues when settings file is not deleted after the migration.