Connecting Reader/Writers to MS SQL Server Instance using Windows Authentication

In order to use Windows Authentication with a Linux/Unix environment, you must use `Kerberos` authentication. By default Ubuntu containers (which Safe uses to build its FME Server containers) do not have the neccessary applications installed to support `Kerberos` authentication. We used the below steps to modify the default FME Engine container to have the required applications for `Kerberos` authentication and create an auto-renewing `Kerberos` ticket for a specified Windows Account are the time the FME Engine container starts.

Check if the MS SQL Server Instance you would like to connect to supports `Kerberos` authentication.

- Your login/users must have rights to read the master database
- The result table results should like `Kerberos` as an `auth_scheme`. If not `Kerberos` authentication should be enabled on the SQL Server or SQL Authentication should be used (does not require the below steps).

```
SELECT auth_scheme FROM sys.dm_exec_connections WHERE session_id=@@spid
```

Create a `kerberos.sh` bash script

- This script will create a `Kerberos` ticket on the FME Engine container
- The `Kerberos` ticket will be auto renewed each day (built in the `k5start` application)
- You can utilize a secret server like Docker Swarm secrets to store and pass the Windows Account username and password to the FME Engine container

```
#!/bin/bash

# Create Kerberos ticket for the specified network user
# used for JDBC connections back to MS SQL Server databases
keytab="/etc/krb5.keytab"
username=$(</<path to Windows Account User Name file>)
password=$(</<path to Windows Account Password file>)
printf "add_entry -password -p $username -k 1 -e arcfour-hmac\n$password\nwrite_kt
$keytab\nread_kt $keytab\nlist\nexit\n" | ktutil
k5start -K 60 -a -f $keytab $username &
```

Modify the existing `start_engine.sh` bash script

- This script originally come from the FME Engine container - `/fmeengine/start_engine.sh`
- Removed reference to checking if the PostgreSQL instance can be reached (this setup uses a MS SQL Server instance for the FME Server Database instead)
- Added reference to execute the `kerberos.sh` script created above before running the `/opt/fmeserver/Server/startEngines.sh` script

```
#!/bin/bash

# Make sure the fmeserverdata localization directory isn't empty
 until [ -n "$(ls -A "/data/fmeserverdata/localization")" ]
 do
    echo "Waiting until shared resources are available..."
    sleep 1
 done

# Run kerberos script to make ticket for Network Login
/fmeengine/kerberos.sh
/opt/fmeserver/Server/startEngines.sh
```

Create FME Engine Dockerfile

- Dockerfile inherits from Safe's `fmeserver-engine:2018-latest` Docker image.
- Installs `krb5-user` and `kstart` applications (for `Kerberos` ticketing)
- Copys `kerberos.sh` and `start_engine.sh` bash scripts to `/fmeengine` directory

```
FROM safesoftware/fmeserver-engine:2018-latest
RUN /bin/bash -c 'apt update'
RUN /bin/bash -c 'DEBIAN_FRONTEND=noninteractive apt-get -y install krb5-user
kstart'
COPY /scripts/kerberos.sh /fmeengine/kerberos.sh
RUN /bin/chmod u+x /fmeengine/kerberos.sh
COPY /scripts/start_engine.sh /fmeengine/start_engine.sh
```

Create new FME Engine Image from the above Dockerfile. Update Safe's Docker Compose file to reference new FME Engine image. Run Docker Compose.

Within FME Server create a new Database Connection

- Type - JDBC
- Connection String is as follows -

```
jdbc:sqlserver://<server/instance>;databaseName=<database>;integratedSecurity=true;
authenticationScheme=JavaKerberos
```

Notes

- When creating a `Kerberos` ticket besure to reference your Windows Account username in the following format - `username@DOMAIN`. The Domain Name should be all caps or the ticket will fail to be created.
- Ensure the Windows Account the `Kerberos` ticket was created for has the approriate permissions for the MS SQL Server Instance you are trying to connect to.

Connecting Reader/Writers to Windows File Share using Windows Authentication

In order to use Windows Authentication with connecting to a Windows Fileshare, the desires shares must be mounted to the Docker Host for FME Server using `SMB`. The below instructions will only work for a `Linux` Docker Host. If you use a `Unix` or `Windows` Docker Host, you will have to share the desires file shares to the FME Engine Container via the Docker settings.

On the Docker Host for FME Server -

Install the following application for `SMB` support

```
apt-get install cifs-utils -y
```

Create `.smbcredentials` file with the following content

- The specified Windows Account should have permissions to the Windows Fileshare

```
username=msusername@Domain
password=mspassword
```

Edit the `/etc/fstab` file (requires `sudo` access), add the following line per file share you wish to connect to

- `vers=2.0` refers to the minimum `SMB` version your infrastructure uses
- If the Docker Hosts restarts, these mounts will be recreated uses the `fstab` file and the `.smbcredentials` file

```
<file share path>    /media/<local path>      cifs    vers=2.0,credentials=<path to
file>/.smbcredentials,iocharset=utf8,file_mode=0777,dir_mode=0777        0       0
```

Run the following command to change the `.smbcredentials` to `Root` (helps protect plain test file from being read by anyone)

```
chmod 600 <path to file>/.smbcredentials
```

Run the following command to mount the file shares references in the `fstab` file to the Host

- Should list out all the host's mounts under if successful

```
mount -a -v
```

Update the Safe FME Server Docker Compose file to add addition `volumes` (bind) to the FME Engine container that reference the file shares on the host

```
fmeserverengine:
  image: safesoftware/fmeserver-engine:2018-latest
  volumes:
   - fmeserver:/data/fmeserverdata
   - /media/<file share>:/media/<file share>
    .
    .
    .
    .
    .
```

Run Docker Compose file to create FME Server Environment

Within FME Server when running a Workspace that makes use of a Windows Fileshare, reference the value of this fileshare path like `/media/<file share>` to use the FME Engine volume mount.