Summary

The goal for the pull_ilo_data.py script is to download system BIOS settings and firmware inventory information from iLO chassis manager. It connects to the iLO using Python iLO REST library, makes query to the system BIOS configurations and firmware inventory, and then downloads and saves the information in JSON file format. The name of the resulting JSON file contains the system model and the iLO name. The script has been tested with Gen9 iLO4 and Gen10 iLO5, in production mode.

Environment and Setup

This script has been tested on Python 3.6.8, but should work with any Python3 version. Pythons can either be installed from OS package manager, such as *apt-get* on Ubuntu, or downloaded from source. Here is a quick start guide from Digital Ocean for instructions to set up a Python3 environment and installing the Pip package manager on Ubuntu 18.04. Dependencies are listed in *requirements.txt*. Use Pip to install the packages

```
operator$ python3 --version
Python 3.6.8
operator$ sudo pip install -r requirements.txt
```

Running the script

The script takes in two parameters, *--iloip* and *--username*, and then it would prompt the operator to enter the login password. It then saves two JSON files to your local directory, one containing the BIOS configurations, and one for the firmware inventory information.

1. --iloip: iLO IP address
2. --username: iLO login username

```
operator$ python pull_ilo_data.py --username administrator --iloip 192.168.0.100
password: iLO login password
>
MXQ63507MJ
Save BIOS settings to
/home/operator/ilorest_pull_configs/ProLiant_DL360_Gen9_iLO4_MXQ63507MJ_bios.json
Save firmware inventory to
/home/operator/ilorest_pull_configs/ProLiant_DL360_Gen9_iLO4_MXQ63507MJ_firmware_inve
ntory.json
```