HUBER+SUHNER
Polatis

Python Scripts for Controlling Polatis All-Optical Switches

Version 1.1

April 2017

1 CONTENTS

2 CREATE OR MOVE CROSS-CONNECTIONS:

For create or move cross-connections, we have to use “create_move_conns.py” python script. You can see the script usage by running “python create_move_conns.py --help” from command prompt like below.

```
D:\polatis_scripts>python create_move_conns.py --help
usage: create_move_conns.py [-h] --host HOST --username USERNAME --password
                            PASSWORD --inports INPORTS --outports OUTPORTS

optional arguments:
  -h, --help           show this help message and exit

required arguments:
  --host HOST                  IP address of the switch
  --username USERNAME  Username
  --password PASSWORD  Password
  --inports INPORTS        'ingPrt' or 'ingPrt1,ingPrt2' or 'ingPrt1-ingPrt2'; eg:
                               1 or 1,2 or 1-5
  --outports OUTPORTS    'egPrt' or 'egPrt1,egPrt2' or 'egPrt1-egPrt2'; eg: 49
                              or 49,50 or 49-53
```

Example for creating cross-connection with single port:

```
D:\polatis_scripts>python create_move_conns.py --host 10.99.99.225 --username admin --password root
--inports 1 --outports 49
2017-03-30 18:15:42,209 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:15:42,835 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:05:01
M  123 COMPLD
;

2017-03-30 18:15:43,141 - pypolatis.crossconnect - INFO - ent-patch::1,49:123:;

2017-03-30 18:15:43,491 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:05:02
M  123 COMPLD
;

2017-03-30 18:15:43,491 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:15:43,807 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:05:02
```

```
M  123 COMPLD
;
D:\polatis_scripts>
```

Example for creating cross-connection with multiple ports:

```
D:\polatis_scripts>python create_move_conns.py --host 10.99.99.225 --username admin --password root
--inports 1,2 --outports 49,50
2017-03-30 18:16:28,945 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:16:29,958 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:05:48
M  123 COMPLD
;

2017-03-30 18:16:30,267 - pypolatis.crossconnect - INFO - ent-patch::1&2,49&50:123:;

2017-03-30 18:16:30,607 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:05:49
M  123 COMPLD
;

2017-03-30 18:16:30,607 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:16:30,930 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:05:49
M  123 COMPLD
;
D:\polatis_scripts>
```

Example for creating cross-connection with port range:

```
D:\polatis_scripts>python create_move_conns.py --host 10.99.99.225 --username admin --password root
--inports 1-3 --outports 49-51
2017-03-30 18:17:41,475 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:17:42,357 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:07:00
M  123 COMPLD
;

2017-03-30 18:17:42,741 - pypolatis.crossconnect - INFO - ent-patch::1&&3,49&&51:123:;

2017-03-30 18:17:43,088 - pypolatis._tl1 - INFO - buff :
   PolatisOXC 19-12-29 02:07:01
M  123 COMPLD
;

2017-03-30 18:17:43,088 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:17:43,407 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:07:01
M  123 COMPLD
;
D:\polatis_scripts>
```

3 RETRIEVE CROSS-CONNECTIONS:

For retrieving connections, we have to use “retrieve_connections.py” python script. You can see the script usage by running “python retrieve_connections.py --help” from command prompt like below.

```
D:\polatis_scripts>python retrieve_connections.py --help
usage: retrieve_connections.py [-h] --host HOST --username USERNAME --password
                               PASSWORD --ports PORTS

optional arguments:
  -h, --help           show this help message and exit

required arguments:
  --host HOST          IP address of the switch
  --username USERNAME  Username
  --password PASSWORD  Password
  --ports PORTS        'prt' or 'prt1,prt2' or 'prt1-prt2' or ALL; eg: 1 or 49,50 or
                        1-53 or ALL (or “” as a stand-in for ALL on older software versions)
```

Please note that on older versions of the Polatis switch software “” (two back-to-back double quotes) must be used instead of ALL to retrieve all switch connections.

Example for retrieving entire connection state with ‘ALL’ keyword:

```
D:\polatis_scripts>python retrieve_connections.py --host 10.99.99.225 --username admin --password
root --ports ALL
2017-03-30 22:18:06,868 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 22:18:07,498 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 06:07:27
M  123 COMPLD
;
```

```
2017-03-30 22:18:07,806 - pypolatis.crossconnect - INFO - rtrv-patch::ALL:123:;

2017-03-30 22:18:08,160 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 06:07:27
M  123 COMPLD
   "1,49"
   “2,50”
   "3,51"
;

2017-03-30 22:18:08,160 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 22:18:08,480 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 06:07:28
M  123 COMPLD
;
```

Example for retrieving cross-connection with single port:

```
D:\polatis_scripts>python retrieve_connections.py --host 10.99.99.225 --username admin --password
root --ports 1
2017-03-30 18:32:09,594 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:32:10,275 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:21:28
M  123 COMPLD
;

2017-03-30 18:32:10,628 - pypolatis.crossconnect - INFO - rtrv-patch::1:123:;

2017-03-30 18:32:11,015 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:21:29
M  123 COMPLD
   "1,49"
;

2017-03-30 18:32:11,015 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:32:11,380 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:21:29
M  123 COMPLD
;
```

Example for retrieving cross-connection with multiple ports:

```
D:\polatis_scripts> python retrieve_connections.py --host 10.99.99.225 --username admin --password root --ports 1,5
2017-03-30 18:38:04,944 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:38:05,590 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:27:24
M  123 COMPLD
;

2017-03-30 18:38:05,904 - pypolatis.crossconnect - INFO - rtrv-patch::1&5:123:;

2017-03-30 18:38:06,240 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:27:24
M  123 COMPLD
   "1,49"
   "5,53"
;

2017-03-30 18:38:06,241 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:38:06,558 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:27:25
M  123 COMPLD
;
```

Example for retrieving cross-connection with port range:

```
D:\polatis_scripts> python retrieve_connections.py --host 10.99.99.225 --username admin --password root --ports 1-5
2017-03-30 18:40:09,412 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:40:10,099 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:29:28
M  123 COMPLD
;

2017-03-30 18:40:10,481 - pypolatis.crossconnect - INFO - rtrv-patch::1&&5:123:;

2017-03-30 18:40:11,014 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:29:29
M  123 COMPLD
   "1,49"
```

```
  "2,50"
  "3,51"
  "5,53"
;

2017-03-30 18:40:11,014 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:40:11,367 - pypolatis._tl1 - INFO - buff :

  PolatisOXC 19-12-29 02:29:29
M  123 COMPLD
;
```

4 DELETE CROSS-CONNECTIONS:

For deleting connections, we have to use “delete_connections.py” python script. You can see the script usage by running “python delete_connections.py --help” from command prompt like below.

```
D:\polatis_scripts> python delete_connections.py --help
usage: delete_connections.py [-h] --host HOST --username USERNAME --password
                    PASSWORD --ports PORTS

optional arguments:
  -h, --help         show this help message and exit

required arguments:
  --host HOST                IP address of the switch
  --username USERNAME  Username
  --password PASSWORD  Password
  --ports PORTS              'prt' or 'prt1,prt2' or 'prt1-prt2 or ALL'; eg: 1 or 49,50 or
                             1-53 or ALL
```

<u>Example for deleting entire connections with ‘ALL’ keyword:</u>

```
D:\polatis_scripts >python delete_connections.py --host 10.99.99.225 --username admin --password
root --ports ALL
2017-03-30 22:40:36,195 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 22:40:37,044 - pypolatis._tl1 - INFO - buff :

  PolatisOXC 19-12-29 06:29:56
M  123 COMPLD
;

2017-03-30 22:40:37,352 - pypolatis.crossconnect - INFO - dlt-patch::ALL:123:;
```

```
2017-03-30 22:40:37,701 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 06:29:57
M  123 COMPLD
;

2017-03-30 22:40:37,701 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 22:40:38,051 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 06:29:57
M  123 COMPLD
;
```

Example for deleting cross-connection with single port:

```
D:\polatis_scripts>python delete_connections.py --host 10.99.99.225 --username admin --password root
--ports 1
2017-03-30 18:42:25,248 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:42:25,918 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:31:44
M  123 COMPLD
;

2017-03-30 18:42:26,225 - pypolatis.crossconnect - INFO - dlt-patch::1:123:;

2017-03-30 18:42:26,566 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:31:45
M  123 COMPLD
;

2017-03-30 18:42:26,566 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:42:26,894 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:31:45
M  123 COMPLD
;
```

Example for deleting cross-connection with multiple ports:

```
D:\polatis_scripts>python delete_connections.py --host 10.99.99.225 --username admin --password root
--ports 1,5
2017-03-30 18:47:42,932 - pypolatis.session - INFO - act-user::admin:123::root;
```

```
2017-03-30 18:47:43,568 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:37:02
M  123 COMPLD
;

2017-03-30 18:47:43,870 - pypolatis.crossconnect - INFO - dlt-patch::1&5:123:;

2017-03-30 18:47:44,215 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:37:02
M  123 COMPLD
;

2017-03-30 18:47:44,215 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:47:44,540 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:37:03
M  123 COMPLD
;
```

Example for deleting cross-connection with port range:

```
D:\polatis_scripts> python delete_connections.py --host 10.99.99.225 --username admin --password root
--ports 1-5
2017-03-30 18:47:24,529 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-30 18:47:25,248 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:36:43
M  123 COMPLD
;

2017-03-30 18:47:25,566 - pypolatis.crossconnect - INFO - dlt-patch::1&&5:123:;

2017-03-30 18:47:25,911 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 19-12-29 02:36:44
M  123 COMPLD
;

2017-03-30 18:47:25,911 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-30 18:47:26,228 - pypolatis._tl1 - INFO - buff :
   PolatisOXC 19-12-29 02:36:44
M  123 COMPLD
;
```

5 EXPORT CROSS-CONNECTIONS:

For exporting cross-connections, we have to use “export_connections.py” python script. You can see the script usage by running “python export_connections.py --help” from command prompt like below.

```
D:\polatis_scripts> python export_connections.py --help
usage: export_connections.py [-h] --host HOST --username USERNAME --password
                             PASSWORD --filename FILENAME

optional arguments:
  -h, --help            show this help message and exit

required arguments:
  --host HOST           IP address of the switch
  --username USERNAME  Username
  --password PASSWORD  Password
  --filename FILENAME   Enter a filename to export(.json will be automatically
                        added); eg: 'conns_state'
```

Example for export operation:

```
D:\polatis_scripts>python export_connections.py --host 10.99.99.225 --username admin --password root
--filename connections

2017-03-30 19:00:00,285 - pypolatis.crossconnect - INFO - Exporting the current connection state in a
file...
2017-03-30 19:00:00,607 - pypolatis.crossconnect - INFO - Export operation completed.
```

JSON file output:

```
{
    "portconns": [
        [
            "1",
            "49"
        ],
        [
            "2",
            "50"
        ],
        [
            "3",
            "51"
        ],
        [
            "4",
            "52"
        ],
        [
            "5",
            "53"
        ]
    ]
}
```

6 IMPORT CROSS-CONNECTIONS:

For importing cross-connections, we have to use “import_connections.py” python script. You can see the script usage by running “python import_connections.py --help” from command prompt like below.

```
D:\polatis_scripts>python import_connections.py --help
usage: import_connections.py [-h] --host HOST --username USERNAME --password
                             PASSWORD --filename FILENAME

optional arguments:
  -h, --help            show this help message and exit

required arguments:
  --host HOST           IP address of the switch
  --username USERNAME   Username
  --password PASSWORD   Password
  --filename FILENAME   Enter your previously exported JSON file; eg:
                        "connections.json"
```

<u>Example for import operation:</u>

```
D:\polatis_scripts>python import_connections.py --host 10.99.99.225 --username admin --password root
--filename connections.json
2017-03-30 19:09:22,598 - pypolatis.crossconnect - INFO - Import operation completed.

D:\polatis_scripts>
```

ERROR HANDLING:

<u>Example for authentication failure:</u>

```
D:\polatis_scripts>python create_move_conns.py --host 10.99.99.225 --username noadmin --password root --inports 1 --outports 49
2017-03-29 15:52:30,331 - pypolatis.session - INFO - act-user::noadmin:123::root;

2017-03-29 15:52:30,458 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 17-03-29 17:22:51
M  123 DENY
   PICC
;

2017-03-29 15:52:30,458 - pypolatis._tl1 - ERROR - Authentication failure. Please check the username or password.
```

Example for invalid OXC port:

```
D:\polatis_scripts>python create_move_conns.py --host 10.99.99.225 --username admin --password root --inports 100 --outports 50
2017-03-29 15:53:37,233 - pypolatis.session - INFO - act-user::admin:123::root;

2017-03-29 15:53:37,855 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 17-03-29 17:23:59
M  123 COMPLD
;

2017-03-29 15:53:37,856 - pypolatis.crossconnect - INFO - ent-patch::100,50:123:;

2017-03-29 15:53:37,877 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 17-03-29 17:23:59
M  123 DENY
   IIAC
   /* Invalid OXC port: 100 */
;

2017-03-29 15:53:37,877 - pypolatis.session - INFO - canc-user::admin:123:;

2017-03-29 15:53:37,897 - pypolatis._tl1 - INFO - buff :

   PolatisOXC 17-03-29 17:23:59
M  123 COMPLD
;
```

Example for invalid file during import operation:

```
D:\polatis_scripts>python import_connections.py --host 10.99.99.225 --username admin --password root
--filename junk_file.txt
2017-03-30 19:14:35,467 - pypolatis.crossconnect - ERROR - Not a json file.Enter  a valid file with json
extension.
D:\polatis_scripts>
```

<u>Example for non-existing file during import operation:</u>

```
D:\polatis_scripts>python import_connections.py --host 10.99.99.225 --username admin --password root
--filename notExist.json
2017-03-30 19:13:42,769 - pypolatis.crossconnect - ERROR - File doesn't exist.

D:\polatis_scripts>
```

<u>Example for file with no connections (while export) during import operation:</u>

```
D:\polatis_scripts>python import_connections.py --host 10.99.99.225 --username a
dmin --password root --filename connections.json
2017-03-30 19:19:22,553 - pypolatis.crossconnect - ERROR - File is empty. Make sure the exported file
has some connections.
D:\polatis_scripts>
```