Requirements
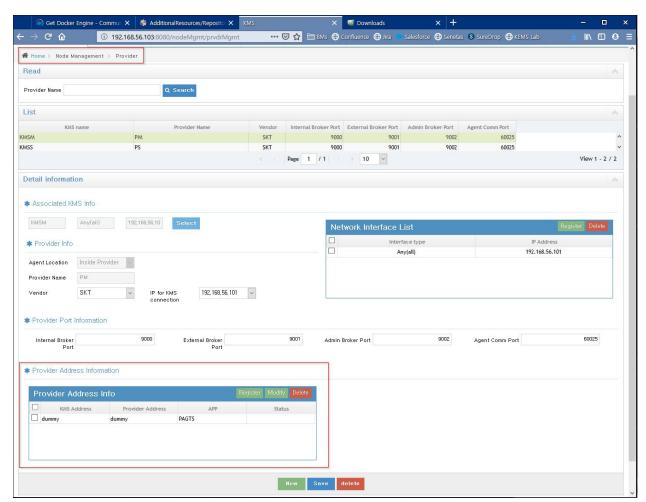
1. python 3.x
2. Selenium 3.141.0 or higher (python lib)
   'pip install selenium'
3. webdriver for Selenium (https://www.seleniumhq.org/download/)
   a. install one of the followings and add to PATH depending on the browser to use.
      - Mozilla GeckoDriver
      - Google Chrome Driver
      - Microsoft Edge Driver
      - Windows IE
4. IDQ KEMS up and running

Features

Add Provider Address

Node Management > Provider → Provider Address Information :
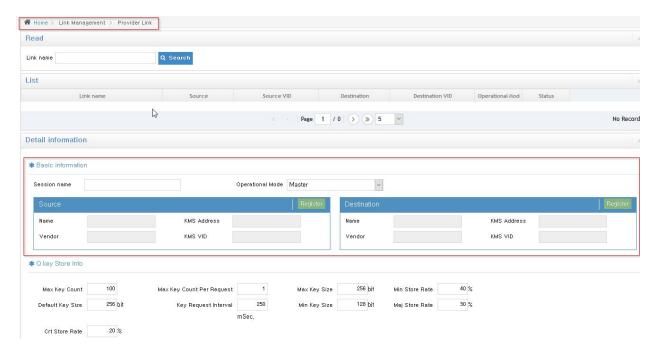
Automation to add QKD-KMS logical sessions for key delivery.

Add Provider Link

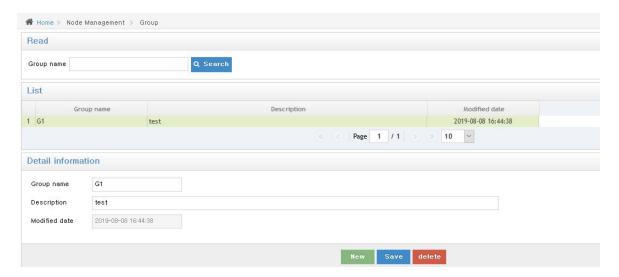Link Management > Provider Link → Detail Information > Basic Information :

Automation to add links between Providers.
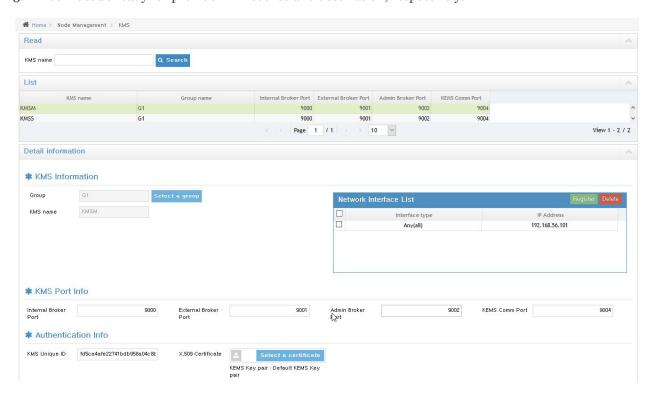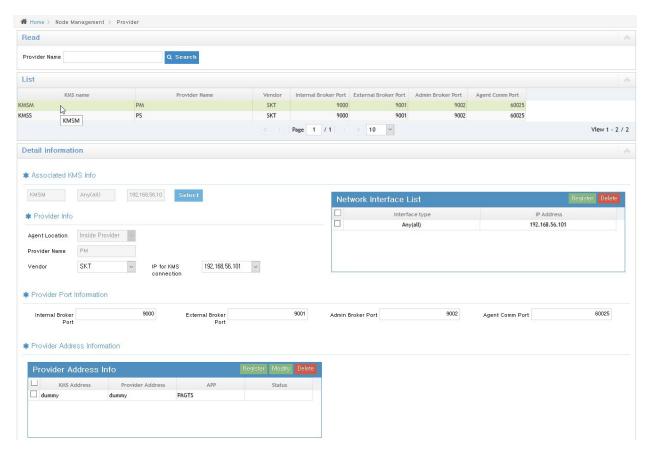

How-to

Prerequisites

1. IDQ KEMS up & running

2. Group must be ready

3. KMSs must be ready for provider link source and destination, respectively.

4. Providers must be ready for provider link source and destination, respectively.

Set up 'config' file

1. copy 'config.template' to 'config' in kems root.

2. update 'config' file in accordance with test environment

* Refer to '[config details](#)' section

Run

'python main.py'

config details

* file format : strict json file plus commenting is possible.

configuration items

server_url

KEMS web server URL

server_account

KEMS admin account id & pwd

node_provider_address

configuration to add logical KMS-QKD connections

```
<format>

"node_provider_address" : {

    <Provider Name> : {

        "op" : <operation>,

        "prv_addr" : <any ID to give to provider>,

        "kms_addr_spec" : {

            "fmt" : <python style format string>,

            "range" : <python style 2-dimension tuple>

        },
```

```
        "prv_agt" : "PAGTS"

    },

        ...

    }
```

1. <Provider Name> : any existing provider name to add addresses to

2. <operation> : one of the following
   - add : to add addresses specified by "prv_addr" & "kms_addr_spec".
   - del : to delete addresses specified by "prv_addr" & "kms_addr_spec".
   - whatever else (e.g. "skip" or "ignore" or "no" …) : to ignore this provider's action for expediency rather than commenting out all the lines for the provider setting.
3. "prv_addr" : any ID as provider end to give to this logical connection between QKD and KMS.
4. "kms_addr_spec" : any ID as KMS end to give to this logical connection between QKD and KMS.
   - "fmt" : python expandable string format composed of constant and variable
     - constant : fixed string in KMS address
     - variable : expandable part in KMS address denoted by '{}' that will be replaced by "range" spec.
   - "range" : python style series of tuples, (start, end). Must end with comma(,) for one entry like "(100,103),".

   e.g. {"fmt" : "101_{}", "range" : "(100,102), (200,201)"} will expand KMS address into 5 instances.

   - 101_100
   - 101_101
   - 101_102
   - 101_200
   - 101_201

5. "prv_agt" : PAGTS only. (other values are future reserved)

link_provider

configuration to add links for registered provider addresses by node_provider_address.

```
<format>

"link_provider" : {

    <connection name format> : {

        "op" : <operation>,

        "conn_name_type" : <name build type>,

        "conn_src" : {

            "prv_name" : <provider name>,

            "kms_addr_spec" : {

                "fmt" : <python expandable string composed of constant
and variable>,

                "range" : <python style series of tuples>

            },

        "conn_dst" : {

            …

        }

    }
```

Read

Link name [   ]   🔍 Search

List

Link name	Source	Source VID	Destination	Destination VID	Operational Mod	Status	

« ‹ Page 1 / 0 › » 5 [∨]   No Record

Detail information

❋ Basic information

Session name [**connection name**]   Operational Mode [Master ∨]

Source [**conn_src**]		Register
Name [**prv_name**]	KMS Address [**kms_addr_spec**]
Vendor [ ]	KMS VID [ ]

Destination [**conn_dst**]		Register
Name [**prv_name**]	KMS Address [**kms_addr_spec**]
Vendor [ ]	KMS VID [ ]

❋ Q key Store Info

Max Key Count [100]	Max Key Count Per Request [1]	Max Key Size [256] bit	Min Store Rate [40] %
Default Key Size [256] bit	Key Request Interval [250] mSec.	Min Key Size [128] bit	Maj Store Rate [30] %

Crt Store Rate [20] %

1. <connection name format> : python string composed of prefix and variable that will decide "connection name" together with "conn_name_type".
   e.g. "`PLink_{}`"
   ○ `PLink_` : prefix
   ○ `{}` : variable
2. "conn_name_type" : how to compose the variable part of <connection name format>.
   ○ order : append ordering number starting from 1.
     e.g. `PLink_1, PLink_2, PLink_3, …`
   ○ combi : append combination of "prv_name" and "kms_address" in the format of
     `<Prefix>_<Src Prv Name>.<KMS addr>:<Dst Prv Name>.<KMS addr>`
     e.g. `PLink_PM.101_100:PS.102_100, PLink_PM.101_101:PS.102_101, …`
3. <operation> : add, del or else (see node_provider_address → <operation> for details)
4. "prv_name" : existing provider name for either source or destination. (see node_provider_address → <Provider Name> for details)
5. "kms_addr_spec" : expandable KMS address spec. (see node_provider_address → "kms_add_spec" for details)

note: conn_src & conn_dst must have one-to-one bijection of address range to complete the connection list.