HSM Keys Configuration Document :

HSM Keys which is provided by Vendor that vendor and we inject in HSM.
After injecting Keys on both side we are able to communicate with each other

We have Different type of Keys:
Like : (Card verification keys,Card Personalization Key,Pin verification Key and **many other**) and all keys have their different name **like** (IMKac is for Card personalize)

Some Keys Configuration are below written:

Usman Mughal (decent.usman143@gmail.com)

HSM KIR Configuration

- Login with Partition Owner
- Command:
  - Login PartitionOwner -Partition part1(Partition name)
  - User 1 = Part1
  - Pass = xxxxxxxxx
  - User 2 = Part2
  - Pass = xxxxxxxxx
- Generate KIR using **command**
  - KeyMgmt generate hsm kir -index 1 -keyLen 2 -clearComp 3 -encryptedComp 0 -variantscheme NV
- Enter **Component** 1
- Check **KVC** if matched then write **yes**
- Enter **Component** 2
- Check **KVC** if matched then write **yes**
- Enter **Component** 3
- Check **KVC** if matched then write **yes**
- Check final **KVC** which is **generated** if **matched** then store the keys

Usman Mughal (decent.usman143@gmail.com)

HSM KMC Configuration

- Login with Partition Owner
- Command:
  - Login PartitionOwner -Partition part1(Partition name)
  - User 1 = Part1
  - Pass = xxxxxxxxx
  - User 2 = Part2
  - Pass = xxxxxxxxx
- Generate KMC using **command**
  - KeyMgmt generate hsm kmc -index 1 -clearComp 3 -encryptedComp 1
- Enter **Component** 1
- Check **KVC** if matched then write **yes**
- Enter **Component** 2
- Check **KVC** if matched then write **yes**
- Enter **Component** 3
- Check **KVC** if matched then write **yes**
- Enter **Encrypted** Key **Component** 1
- Ignore Encrypted Key kvc if not matched and Write yes
- Check final **KVC** which is **generated** if **matched** then store the keys

Usman Mughal (decent.usman143@gmail.com)

HSM IMK-AC Configuration

- Login with Partition Owner
- Command:
  - Login PartitionOwner -Partition part1(Partition name)
  - User 1 = Part1
  - Pass = xxxxxxxxx
  - User 2 = Part2
  - Pass = xxxxxxxxx
- Generate imkac using **command**
  - KeyMgmt generate hsm imkac -index 1 -clearComp 2 -encryptedComp 0 -algo DES -keyBit 128 -kvcAlgo ZL6
- Enter **Component** 1
- Check **KVC** if matched then write **yes**
- Enter **Component** 2
- Check **KVC** if matched then write **yes**
- Check final **KVC** which is **generated** if **matched** then store the keys

Usman Mughal (decent.usman143@gmail.com)

HSM IMK-SMI Configuration

- Login with Partition Owner
- Command:
  - Login PartitionOwner -Partition part1(Partition name)
  - User 1 = Part1
  - Pass = xxxxxxxxx
  - User 2 = Part2
  - Pass = xxxxxxxxx
- Generate imksmi using **command**
  - KeyMgmt generate hsm imksmi -index 1 -clearComp 2 -encryptedComp 0 -algo DES -keyBit 128 -kvcAlgo ZL6
- Enter **Component** 1
- Check **KVC** if matched then write **yes**
- Enter **Component** 2
- Check **KVC** if matched then write **yes**
- Check final **KVC** which is **generated** if **matched** then store the keys

HSM IMK-SMC Configuration

- Login with Partition Owner
- Command:
  - Login PartitionOwner -Partition part1(Partition name)
  - User 1 = Part1
  - Pass = xxxxxxxxx
  - User 2 = Part2
  - Pass = xxxxxxxxx
- Generate imksmc using **command**
  - KeyMgmt generate hsm imksmc -index 1 -clearComp 2 -encryptedComp 0 -algo DES -keyBit 128 -kvcAlgo ZL6
- Enter **Component** 1
- Check **KVC** if matched then write **yes**
- Enter **Component** 2
- Check **KVC** if matched then write **yes**
- Check final **KVC** which is **generated** if **matched** then store the keys

HSM IMK-DAC Configuration

- Login with Partition Owner
- Command:
  - Login PartitionOwner -Partition part1(Partition name)
  - User 1 = Part1
  - Pass = xxxxxxxxx
  - User 2 = Part2
  - Pass = xxxxxxxxx
- Generate imkdac using **command**
  - KeyMgmt generate hsm imkdac -index 1 -clearComp 2 -encryptedComp 0 -algo DES -keyBit 128 -kvcAlgo ZL6
- Enter **Component** 1
- Check **KVC** if matched then write **yes**
- Enter **Component** 2
- Check **KVC** if matched then write **yes**
- Check final **KVC** which is **generated** if **matched** then store the keys