

# Cosmian KMS Source Code Modification - 002

## Implement the REST API to Request EnrolData to the KMS/Cosmian

---

### Code Modifications

#### Modification 01:

Folder: `crate/server/src/enrol_data`

`enrol_data` folder includes all newly defined content to fulfill the requirements of `GenerateEnrolData` and `GetEnrolData`.

#### Modification 02:

File: `crate/server/src/lib.rs`

Include `enrol_data` in `lib.rs` to make `enrol_data` a publicly available module. So then it can be accessible anywhere inside the relevant crate. [`crate/server/src`]

Add,

```
pub mod enrol_data;
```

#### Modification 03:

File: `crate/server/src/routes/kmip.rs`

Import required functions defined at the `enrol_data`.

```
use crate::enrol_data::{generate_enrol_data::handle_generate_enroldata,  
get_enrol_data::handle_get_enroldata};
```

#### Modification 04:

File: `crate/server/src/routes/kmip.rs`

Replace:

```
let ttlv = handle_ttlv(&kms, &ttl, &user, database_params.as_ref()).await?;  
Ok(Json(ttlv))
```

Using:

```
let ttlv_out: TTLV;

    if ttlv.tag.as_str() == "GenerateEnrolData" {
        ttlv_out =
            handle_generate_enroldata(&kms, &body, &ttl, &user,
database_params.as_ref()).await?;
    } else if ttlv.tag.as_str() == "GetEnrolData" {
        ttlv_out =
            handle_get_enroldata(&kms, &body, &ttl, &user, database_params.as_ref()).await?;
    } else {
        /*Note: All the operations happens here and receive the response */
        ttlv_out = handle_ttlv(&kms, &ttl, &user, database_params.as_ref()).await?;
    }
    Ok(Json(ttlv_out))
```

## Code Compilation

Run the Actix web application, Specifically:

- **Rust:** Install it using rustup if it's not already installed.
- **Cargo:** This comes with the Rust installation.

```
sudo apt update
sudo apt install curl
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
source $HOME/.cargo/env
cargo --version
```

To run the server, the following command can be used. [at the project root directory]

```
cargo run --bin cosmian_kms_server
```

The server will be exposed to the default port 9998 of the local PC.

**If you want to build a docker image and run it in a container:**

```
docker build -t cosmian_kms .
docker run -d -p 9998:9998 --name kms cosmian_kms

docker start kms
docker stop kms
```

## Testing

We can test the API with the specific JSON TTLV POST requests by sending them to the `http://localhost:9998/kmip/2_1` endpoint.

### Using CURL:

#### GenerateEnrolData\_curl\_command

```
curl -v -X POST -H "Content-Type: application/json" -d '{
  "tag": "GenerateEnrolData",
  "type": "Structure",
  "value": [
    {
      "tag": "SerialNumber",
      "type": "Enumeration",
      "value": "SN00001234"
    },
    {
      "tag": "PartNumber",
      "type": "Enumeration",
      "value": "PN00001234"
    }
  ]
}' http://0.0.0.0:9998/kmip/2_1
```

#### GetEnrolData\_curl\_command

```
curl -v -X POST -H "Content-Type: application/json" -d '{
  "tag": "GetEnrolData",
  "type": "Structure",
  "value": [
    {
      "tag": "UUID",
      "type": "Enumeration",
      "value": "<add-the-relevant-uuid>"
    }
  ]
}' http://0.0.0.0:9998/kmip/2_1
```

---

-End of the Document-