**Enrollment APIs**

It provides functionality to enroll directly without use of registration client.

**APIs:**

1. **Create packet:**  
   POST**:** {{url}}/commons/v1/packetmanager/createPacket  
   Packet manager api to create packet and upload to packet store.  
   Request Sample:

|  |
| --- |
| {  "id": "string",  "version": "string",  "requesttime": "2019-08-24T14:15:22Z",  "metadata": {},  "request": {  "id": "string",  "additionalInfoReqId": "string",  "refId": "string",  "offlineMode": true,  "process": "string",  "source": "string",  "schemaVersion": "string",  "schemaJson": "string",  "fields": {  "property1": "string",  "property2": "string"  },  "metaInfo": {  "property1": "string",  "property2": "string"  },  "documents": {  "property1": {  "document": [  "string"  ],  "value": "string",  "type": "string",  "format": "string",  "refNumber": "string"  },  "property2": {  "document": [  "string"  ],  "value": "string",  "type": "string",  "format": "string",  "refNumber": "string"  }  },  "audits": [  {  "property1": "string",  "property2": "string"  }  ],  "biometrics": {  "property1": {  "version": {  "major": 0,  "minor": 0  },  "cbeffversion": {  "major": 0,  "minor": 0  },  "birInfo": {  "creator": "string",  "index": "string",  "payload": [  "string"  ],  "integrity": true,  "creationDate": "2019-08-24T14:15:22Z",  "notValidBefore": "2019-08-24T14:15:22Z",  "notValidAfter": "2019-08-24T14:15:22Z"  },  "segments": [  {  "version": {  "major": 0,  "minor": 0  },  "cbeffversion": {  "major": 0,  "minor": 0  },  "birInfo": {  "creator": "string",  "index": "string",  "payload": [  "string"  ],  "integrity": true,  "creationDate": "2019-08-24T14:15:22Z",  "notValidBefore": "2019-08-24T14:15:22Z",  "notValidAfter": "2019-08-24T14:15:22Z"  },  "bdbInfo": {  "challengeResponse": [  "string"  ],  "index": "string",  "format": {  "organization": "string",  "type": "string"  },  "encryption": true,  "creationDate": "2019-08-24T14:15:22Z",  "notValidBefore": "2019-08-24T14:15:22Z",  "notValidAfter": "2019-08-24T14:15:22Z",  "type": [  "SCENT"  ],  "subtype": [  "string"  ],  "level": "RAW",  "product": {  "organization": "string",  "type": "string"  },  "captureDevice": {  "organization": "string",  "type": "string"  },  "featureExtractionAlgorithm": {  "organization": "string",  "type": "string"  },  "comparisonAlgorithm": {  "organization": "string",  "type": "string"  },  "compressionAlgorithm": {  "organization": "string",  "type": "string"  },  "purpose": "VERIFY",  "quality": {  "algorithm": {  "organization": "string",  "type": "string"  },  "score": 0,  "qualityCalculationFailed": "string"  }  },  "bdb": [  "string"  ],  "sb": [  "string"  ],  "birs": [  {}  ],  "sbInfo": {  "format": {  "organization": "string",  "type": "string"  }  },  "others": {  "property1": "string",  "property2": "string"  }  }  ],  "others": {  "property1": "string",  "property2": "string"  }  },  "property2": {  "version": {  "major": 0,  "minor": 0  },  "cbeffversion": {  "major": 0,  "minor": 0  },  "birInfo": {  "creator": "string",  "index": "string",  "payload": [  "string"  ],  "integrity": true,  "creationDate": "2019-08-24T14:15:22Z",  "notValidBefore": "2019-08-24T14:15:22Z",  "notValidAfter": "2019-08-24T14:15:22Z"  },  "segments": [  {  "version": {  "major": 0,  "minor": 0  },  "cbeffversion": {  "major": 0,  "minor": 0  },  "birInfo": {  "creator": "string",  "index": "string",  "payload": [  "string"  ],  "integrity": true,  "creationDate": "2019-08-24T14:15:22Z",  "notValidBefore": "2019-08-24T14:15:22Z",  "notValidAfter": "2019-08-24T14:15:22Z"  },  "bdbInfo": {  "challengeResponse": [  "string"  ],  "index": "string",  "format": {  "organization": "string",  "type": "string"  },  "encryption": true,  "creationDate": "2019-08-24T14:15:22Z",  "notValidBefore": "2019-08-24T14:15:22Z",  "notValidAfter": "2019-08-24T14:15:22Z",  "type": [  "SCENT"  ],  "subtype": [  "string"  ],  "level": "RAW",  "product": {  "organization": "string",  "type": "string"  },  "captureDevice": {  "organization": "string",  "type": "string"  },  "featureExtractionAlgorithm": {  "organization": "string",  "type": "string"  },  "comparisonAlgorithm": {  "organization": "string",  "type": "string"  },  "compressionAlgorithm": {  "organization": "string",  "type": "string"  },  "purpose": "VERIFY",  "quality": {  "algorithm": {  "organization": "string",  "type": "string"  },  "score": 0,  "qualityCalculationFailed": "string"  }  },  "bdb": [  "string"  ],  "sb": [  "string"  ],  "birs": [  {}  ],  "sbInfo": {  "format": {  "organization": "string",  "type": "string"  }  },  "others": {  "property1": "string",  "property2": "string"  }  }  ],  "others": {  "property1": "string",  "property2": "string"  }  }  }  }  } |

Response Sample:

|  |
| --- |
| {  "id": "string",  "version": "string",  "responsetime": "2019-08-24T14:15:22Z",  "metadata": {},  "response": [  {  "id": "string",  "packetName": "string",  "source": "string",  "process": "string",  "refId": "string",  "schemaVersion": "string",  "signature": "string",  "encryptedHash": "string",  "providerName": "string",  "providerVersion": "string",  "creationDate": "string"  }  ],  "errors": [  {  "errorCode": "string",  "message": "string"  }  ]  } |

{

1. **Workflow instance:**  
   POST**:** {{url}}/registrationprocessor/v1/workflowmanager/workflowinstance  
   Registration processor api to trigger the workflow.  
   Request:

|  |
| --- |
| {  "id": "string",  "version": "string",  "requesttime": "2019-08-24T14:15:22Z",  "metadata": { },  "request": {  "registrationId": "string",  "process": "string",  "source": "string",  "additionalInfoReqId": "string"  }  } |

Sample request:

|  |
| --- |
| {  "id": "mosip.registration.processor.workflow.create",  "version": "1.0",  "requesttime": "2019-08-24T14:15:22Z",  "request": {  "registrationId": "10001104360003820230721101145",  "process": "NEW",  "source": "REGISTRATION\_CLIENT",  "additionalInfoReqId": "string"  }  } |

Response Sample:

|  |
| --- |
| {  "id": " mosip.registration.processor.workflow.create",  "version": "1.0",  "responsetime": "2019-08-24T14:15:22Z",  "response": {  "workflowInstanceId": "d7c2c629-dd03-4caa-a63c-f6c9af7fb1d7"  },  "errors": [  {  "errorCode": "string",  "message": "string"  }  ]  } |

**Flow:**

1. Create packet api to create and store packet in packet store.
2. Take registration id from the response and use in the workflow instance api.
3. This api will populate regprc.registration\_list table.
4. Also regprc.registration table with status code as RESUMABLE and beginning stage name which is configurable(**config name- mosip.regproc.workflow-manager.instance.beginning.stage**)
5. RESUMABLE records are picked up by reprocessor which will start the process from respective stage.