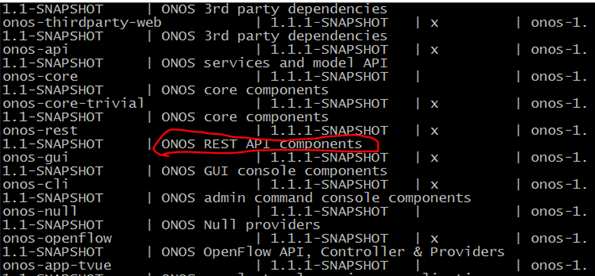
**ONOS Rest API analysis**

After installing ONOS on the server, we can check all features of ONOS through “**feature:list**” command. If ONOS installed properly, you can see REST API in that list.



ONOS feature list

On the ONOS console, execute following command for enabling REST API.

1. feature:install onos-rest

Now we can use ONOS Rest API. If you generated a network, and connected it to the ONOS controller, you can directly check its REST API using following.

**Please note that the all endpoints need to be prefixed with: http://<controller IP>:8181/onos/v1/**

**Example:** <http://127.0.0.1:8181/onos/v1/devices> if ONOS is running on your local machine.

Moreover, instead of devices on above example, you can put hosts, links and so on.

The entries in **bold** are currently implemented. You can understand GET is equal to **http://<controller IP>:8181/onos/v1/ prefix.**

**Device**

|  |  |
| --- | --- |
| **GET /devices** | **Lists all infrastructure devices.** |
| **GET /devices/{deviceId}** | **Lists details of a specific infrastructure device.** |
| **GET /devices/{deviceId}/ports** | **Lists ports of a specific infrastructure device.** |
| POST /devices | Creates a new infrastructure device into inventory (future). |
| PUT /devices/{deviceId} | Updates a device - attributes, availability, mastership. |
| DELETE /devices/{deviceId} | Deletes an infrastructure device from inventory. |

**Link**

|  |  |
| --- | --- |
| **GET /links** | **Lists all infrastructure links.** |
| **GET /links?{device=deviceId}{port=portNumber}{direction=[ALL,INGRESS,EGRESS]}** | **Lists details of a link.** |
| POST /links | Creates a new infrastructure link into inventory (future). |
| PUT /links/{linkId} | Updates a link - attributes, e.g. type. |
| DELETE /link/{linkId} | Deletes an infrastructure link from inventory. |

**Host**

|  |  |
| --- | --- |
| **GET /hosts** | **Lists all end-stations hosts.** |
| **GET /hosts/{hostId}** | **Lists details of a specific end-station host specified by the host ID.** |
| **GET /hosts/{mac}/{vlan}** | **Lists details of a specific end-station host specified by the host MAC address and VLan Id.** |
| POST /hosts | Creates a new end-station host into inventory (future). |
| PUT /hosts/{hostId} | Updates an end-station host - attributes, e.g. IP, location. |
| DELETE /hosts/{hostId} | Deletes an end-station host from inventory. |

**Topology**

|  |  |
| --- | --- |
| **GET /topology** | Gets overview of the current topology. |
| **GET /topology/clusters** | Gets list of topology cluster overviews. |
| **GET /topology/clusters/{clusterId}** | Gets overview of a specific topology cluster. |
| **GET /topology/clusters/{clusterId}/devices** | Gets infrastructure devices that belong to the specified topology cluster. |
| **GET /topology/clusters/{clusterId}/links** | Gets infrastructure links that belong to the specified topology cluster. |
| **GET /topology/broadcast/{connectPoint}** | Gets indication whether the given point permits broadcast. |
| **GET /topology/infrastructure/{connectPoint}** | Gets indication whether the given point belongs to network infrastructure. |

**Path**

|  |  |
| --- | --- |
| **GET /paths/{elementId}/{elementId}** | Gets set of pre-computed shortest paths between the specified source and destination network elements. |

**Flow**

|  |  |
| --- | --- |
| **GET /flows/{deviceId}** | **Gets list of flow rules applied to the specified infrastructure device.** |
| **GET /flows/{deviceId}/{flowId}** | **Gets details of a specified flow rule.** |
| **GET /flows** | **Gets details of all flow rules in the system.** |
| POST /flows/ | Applies batch of flow rule operations; this includes add/remove operations. |
| POST /flows/{deviceId} | Applies a new flow rule on the specified device. |
| DELETE /flows/{deviceId}/{flowId} | Deletes a flow rule from the specified device. |
| GET /flows/link/{linkId} | Gets aggregate statistics for all flows traversing the given link. |

**Intent**

|  |  |
| --- | --- |
| **GET /intents/{intent-id}** | **Gets the details for the given Intent object.** |
| **GET /intents** | **Gets a collection containing all the Intent objects currently in the system.** |
| POST /intents | Creates a new Intent object. |
| POST /intents | Creates a batch of Intent objects (pending -may not be included in finalized API). |
| DELETE /intent/{Intent-id} | Removes an intent from the system. |

**Application**

|  |  |
| --- | --- |
| **GET /applications** | **Gets a list of all installed applications.** |
| **GET /applications/{app-name}** | **Gets information about the named application.** |
| **POST /applications/** | **Installs application using the posted *app.xml* or application package file (ZIP).** |
| **DELETE /applications/{app-name}** | **Uninstalls the named application.** |
| **POST /applications/{app-name}/active** | **Activates the named application.** |
| **DELETE /applications/{app-name}/active** | **Deactivates the named application.** |

References

1. <https://wiki.onosproject.org/pages/viewpage.action?pageId=1048699>
2. <https://groups.google.com/a/onosproject.org/forum/#!topic/onos-deploy/9K6KgHU5TUI>