

Published

2023-05-22

REST API Orchestration Guide

for Paragon Insights

# **Introduction**

This documentation describes how to integrate Paragon Insights with a network service orchestrator via the Paragon Insights REST API. This is a RESTful API that adheres to the standardized OpenAPI/Swagger 2.0 definition format, thereby enabling easy and efficient orchestration of Paragon Insights operations.

# **Conventions**

The following abbreviations are used in this document:

|  |  |
| --- | --- |
| **Abbreviation** | **Meaning** |
| REST | REpresentational State Transfer |
| DG | Device Group |
| NG | Network Group |
| IAM | Identity and Access Management |

# **Overview**

Paragon Insights is a highly automated and programmable device-level diagnostics and network analytics tool that provides consistent and coherent operational intelligence across network deployments. Paragon Insights integrates multiple data collection methods (such as Junos Telemetry Interface (JTI), NETCONF, syslog, and SNMP) to aggregate and correlate large volumes of time-sensitive telemetry data, thereby providing a multidimensional and predictive view of the network. Additionally, Paragon Insights translates troubleshooting, maintenance, and real-time analytics into an intuitive user experience to give network operators actionable insights into the health of individual devices and of the overall network.

Paragon Insights offers closed-loop automation. The automation workflow can be divided into following steps.

Paragon Insights Closed-Loop Automation
Workflow

1. Define—The user defines the health parameters of key network elements through customizable key performance indicators (KPIs), rules, and playbooks, by using the tools provided by Paragon Insights.
2. Collect—Paragon Insights collects rule-based telemetry data from multiple devices using the collection methods specified for the different network devices.
3. Store—Paragon Insights stores time-sensitive telemetry data in a time-series database (TSDB). This allows users to query, perform operations on, and write new data back to the database, days, or even weeks after the initial storage.
4. Analyze—Paragon Insights analyzes telemetry data based on the specified KPIs, rules, and playbooks.
5. Visualize—Paragon Insights provides multiple ways for you to visualize the aggregated telemetry data through its web-based GUI to gain actionable and predictive insights into the health of your devices and the overall network.
6. Notify—Paragon Insights notifies you through the GUI and notification alarms when problems in individual devices or in the network are detected.
7. Act—Paragon Insights performs user-defined actions to help resolve and proactively prevent network problems.

# **Identity Management**

The authentication of user is defined through following entities:

* Roles : This is a collection of capabilities. A capability represent the service and method associated with it
* Scopes: It provides a context for assigning roles to user. User is assigned role on a scope that will be used for RBAC enforcement.

Scopes can be hierarchical, for example:

* + Service Provider (this is default scope for Paragon Insights )
  + Operating Company (OpCo)
  + Tenant
  + Department
* Users : An account created with Paragon. The user can be associated with multiple scope and the access is controlled thorough the rules mapped to that scope.

The authentication works using these concept

1. User login through registered credentials, it returns a list of scopes and a token for next step
2. User can now try opening a login session with any one of the available scope (scope\_id) and token provided
3. Upon successful authentication it will return the available capabilities for that session along with token (id\_token).
4. The id\_token will be used for any further API call.

# **Managing through REST API**

The above workflow can be managed through REST APIs. Each REST API operation consists of an HTTP method (for example, GET, PUT, POST, or DELETE) applied to a URL with a specified syntax. The operations are divided into two categories:

|  |  |
| --- | --- |
| Path | Description |
| {protocol}://{host IP}/api/v2/config/{} | Any API path after this endpoint refers to the configurations management |
| {protocol}://{host IP}/api/v2/ | Everything else (except starting from config) would be used to retrieve health data and related information |

## Authentication

All the APIs are protected with IAM policies. To access the API the user should generate an authentication token using a valid credential. Here are the steps to obtain a valid token.

1. Authenticate to fetch the user scope.

**URL** : {protocol}://{host IP}/iam/authenticate

**Method** : POST

**Data example** All fields must be sent.

|  |
| --- |
| {  “user”: {  “name”: “spadmin”  },  “password”: “changeme”,  “methods”: [  “PASSWORD”  ]  } |

**Success Response**

**Code** : 200 OK

**Content example**

|  |
| --- |
| {  “status”: “AUTHENTICATED”,  “user\_id”: “0fcaf17d-f840-46f3-8374-fca06f567775”,  “id\_token”: “id token with basic authentication information in jwt token format”,  “token”: {  “user\_id”: “0fcaf17d-f840-46f3-8374-fca06f567775”,  “user\_name”: “spadmin”,  “iat”: “1606389958”,  “exp”: “1606393558”  },  “scopes”: [  {  “id”: “d020fba7-10e0-4d41-a474-75347bee0fd0”,  “name”: “default”,  “type”: “SP”  },  {  “id”: “4707bebc-c554-46da-9966-6e5b848a8b44”,  “name”: “tenant1”,  “type”: “Tenant”,  “parent\_id”: “d020fba7-10e0-4d41-a474-75347bee0fd0”,  “parent\_name”: “default”,  “parent\_type”: “SP”  }  ]  } |

**Curl Example:**

|  |
| --- |
| # curl 'https://<Paragon Host>/iam/authenticate' \  > -H 'accept: \*/\*' \  > -H 'accept-language: en-US,en;q=0.9' \  > -H 'content-type: application/json' \  > --data-raw $'{"user":{"domain":"default","name":"admin"},"password":"XXXXXX","methods":["PASSWORD"]}' \  > --compressed \  > --insecure  {  "status": "AUTHENTICATED",  "user\_id": "3fdff8df-7eac-4f4b-b2ec-e8a838ee21f3",  "id\_token": "eyJhbGciOiJSUzI1N…",  "token": {  "user\_id": "3fdff8df-7eac-4f4b-b2ec-e8a838ee21f3",  "user\_name": "admin",  "iat": "1687517579",  "exp": "1687521179",  "unique\_identifier": "b1fcf6c7-cb82-4300-a53e-db2a42aec87c-1687517579",  "identity\_provider\_type": "Local"  },  "scopes": [  {  "id": "eecda7d9-d021-4665-9fe9-05348437c091",  "name": "default",  "display\_name": "default",  "type": "SP"  },  {  "id": "2069c55f-6b83-4941-8f6a-ee0ebdc15ea9",  "name": "northstar",  "display\_name": "northstar",  "type": "Tenant",  "parent\_id": "eecda7d9-d021-4665-9fe9-05348437c091",  "parent\_name": "default",  "parent\_type": "SP"  },  {  "id": "7c211b2b-2df2-4a0e-833c-2beb90623641",  "name": "tm",  "display\_name": "tm",  "type": "Tenant",  "parent\_id": "eecda7d9-d021-4665-9fe9-05348437c091",  "parent\_name": "default",  "parent\_type": "SP"  },  {  "id": "ada7dbe0-e662-4b1d-ab92-e5310b37c19e",  "name": "ems-dmon",  "display\_name": "ems-dmon",  "type": "Tenant",  "parent\_id": "eecda7d9-d021-4665-9fe9-05348437c091",  "parent\_name": "default",  "parent\_type": "SP"  }  ],  "identity\_provider\_type": "Local"  } |

1. Authenticate with scope to get id token

**URL :** {protocol}://{host IP}/iam/authenticate

**Method** : POST

Data example All fields must be sent.

{{id\_token/access\_token}} will be the token from the authenticate api call which has no scope.

{{scope\_id}} will be the scopeID for a specific tenant/scope from the authenticate api call .

|  |
| --- |
| {  “token”: “{{id\_token/access\_token}}” ,  “methods”: [  “TOKEN”  ],  “scope\_id”: “{{scope\_id}}”  } |

**Success Response**

**Code** : 200 OK

Content example

|  |
| --- |
| {  “status”: “AUTHENTICATED”,  “user\_id”: “0fcaf17d-f840-46f3-8374-fca06f567775”,  “id\_token”: “id token with basic authentication information in jwt token format”,  “token”: {  “user\_id”: “0fcaf17d-f840-46f3-8374-fca06f567775”,  “user\_name”: “tenant1@juniper.net”,  “iat”: “1597203830”,  “exp”: “1597207430”,  “roles”: [  {  “id”: “187fc3e7092546a2b22bedb3c80a9a3e”,  “name”: “tenant-admin”,  “type”: “Tenant”  }  ],  “intents”: [  “Roles:C”,  “Roles:D”,  “Roles:R”,  “Roles:U”,  “Users:C”,  “Users:D”,  “Users:R”,  “Users:U”  ],  “scope”: {  “id”: “57db70d64d3e4ca383ea98ea03dcf712”,  “name”: “tenant1”,  “type”: “Tenant”  },  “parent\_scopes”: [  {  “id”: “d020fba7-10e0-4d41-a474-75347bee0fd0”,  “name”: “default”,  “type”: “SP”  }  ],  “capabilities”: [  {  “service”: “iam.IamService”,  “methods”: [  “CreateRole”,  “DeleteRole”,  “DeleteUser”,  “GetRole”,  “GetScope”,  “GetScopeGroup”,  “GetUser”,  “ListRole”,  “ListRoleassignment”,  “ListScope”,  “ListScopeGroup”,  “ListUser”,  “UpdateRole”  ]  },  {  “service”: “iam.IamRPCService”,  “methods”: [  “CreateUser”,  “EditUser”,  “GetCapabilityCatalog”  ]  }  ]  }  } |

Curl Example:

|  |
| --- |
| # curl -X POST 'https://<Paragon Host>/iam/authenticate' \  > -H 'content-type: application/json' \  > --data-raw '{"token":"eyJhbGciOiJS…","methods":["TOKEN"],"scope\_id":"eecda7d9-d021-4665-9fe9-05348437c091"}' \  > --compressed \  > --insecure  {  "status": "AUTHENTICATED",  "user\_id": "3fdff8df-7eac-4f4b-b2ec-e8a838ee21f3",  "id\_token": "eyJhbGciOiJS…",  "token": {  "user\_id": "3fdff8df-7eac-4f4b-b2ec-e8a838ee21f3",  "user\_name": "admin",  "iat": "1687518046",  "exp": "1687521646",  "roles": [  {  "id": "46b63896-bb40-45df-9560-3d40438d8527",  "name": "sp-admin",  "type": "SP",  "predefined": true  }  ],  "intents": [  "ATOM:R",  "Alarm Management:C",  "Alarm Management:D",  "Alarm Management:R",  "Alarm Management:U",  "Configuration Templates:C",  "Configuration Templates:D",  "Configuration Templates:Deploy Config",  "Configuration Templates:R",  "Configuration Templates:U",  "Device Connectivty Edge Service:R",  "Device Families:C",  "Device Families:D",  "Device Families:R",  "Device Families:U",  "Device Images:C",  "Device Images:D",  "Device Images:Deploy Image",  "Device Images:Deploy License",  "Device Images:Deploy Local Certificate",  "Device Images:R",  "Device Images:Stage Image",  "Device Images:U",  "Device Management:C",  "Device Management:D",  "Device Management:R",  "Device Management:U",  "Device Model:C",  "Device Model:D",  "Device Model:R",  "Device Model:U",  "Device Templates:C",  "Device Templates:D",  "Device Templates:R",  "Device Templates:U",  "Devicemodel Connector:C",  "Dmon Proxy:R",  "EmailTemplates:C",  "EmailTemplates:D",  "EmailTemplates:R",  "EmailTemplates:U",  "Export:R",  "External Application Configuration:C",  "External Application Configuration:D",  "External Application Configuration:R",  "External Application Configuration:U",  "IdentityProviderPriority:R",  "IdentityProviderPriority:U",  "Import:C",  "Jobs Purge:C",  "Jobs Purge:D",  "Jobs Purge:R",  "Jobs Purge:U",  "Jobs:C",  "Jobs:R",  "LDAP:C",  "LDAP:R",  "LDAP:U",  "Logs:C",  "Logs:D",  "Logs:R",  "OIDC:C",  "OIDC:D",  "OIDC:R",  "OIDC:U",  "OpCos:C",  "OpCos:D",  "OpCos:R",  "Roles:C",  "Roles:D",  "Roles:R",  "Roles:U",  "SAML:C",  "SAML:D",  "SAML:R",  "SAML:U",  "SMTP:C",  "SMTP:R",  "SMTP:U",  "Telemetry:R",  "Telemetry:W",  "Tenants:C",  "Tenants:D",  "Tenants:R",  "Tenants:U",  "User Groups:C",  "User Groups:D",  "User Groups:R",  "User Groups:U",  "Users:C",  "Users:D",  "Users:R",  "Users:Send Activation Mail",  "Users:U",  "Workflow Definition:C",  "Workflow Definition:D",  "Workflow Definition:R",  "Workflow Definition:U",  "authentication-settings:C",  "authentication-settings:R",  "authentication-settings:U",  "hb-administration-backup-destinations:R",  "hb-administration-backup-devicegroups:R",  "hb-administration-backup-devices:R",  "hb-administration-backup-helperfile:R",  "hb-administration-backup-ingest-settings:R",  "hb-administration-backup-networkgroups:R",  "hb-administration-backup-notification-settings:R",  "hb-administration-backup-playbooks:R",  "hb-administration-backup-plots:R",  "hb-administration-backup-profile-settings:R",  "hb-administration-backup-reports:R",  "hb-administration-backup-retention-policies:R",  "hb-administration-backup-schedulers:R",  "hb-administration-backup-topics:R",  "hb-administration-backup-workflows:R",  "hb-administration-debug:R",  "hb-administration-log-collection:R",  "hb-administration-restore-destinations:C",  "hb-administration-restore-devicegroups:C",  "hb-administration-restore-devices:C",  "hb-administration-restore-helperfile:C",  "hb-administration-restore-ingest-settings:C",  "hb-administration-restore-networkgroups:C",  "hb-administration-restore-notification-settings:C",  "hb-administration-restore-playbooks:C",  "hb-administration-restore-plots:C",  "hb-administration-restore-profile-settings:C",  "hb-administration-restore-reports:C",  "hb-administration-restore-retention-policies:C",  "hb-administration-restore-schedulers:C",  "hb-administration-restore-topics:C",  "hb-administration-restore-workflows:C",  "hb-configuration-device:C",  "hb-configuration-device:D",  "hb-configuration-device:R",  "hb-configuration-device:U",  "hb-configuration-devicegroup:C",  "hb-configuration-devicegroup:D",  "hb-configuration-devicegroup:R",  "hb-configuration-devicegroup:U",  "hb-configuration-network:C",  "hb-configuration-network:D",  "hb-configuration-network:R",  "hb-configuration-network:U",  "hb-configuration-playbooks:C",  "hb-configuration-playbooks:D",  "hb-configuration-playbooks:Execute",  "hb-configuration-playbooks:R",  "hb-configuration-playbooks:Rule Editor",  "hb-configuration-playbooks:U",  "hb-configuration-playbooks:Upload",  "hb-configuration-rca:C",  "hb-configuration-rca:D",  "hb-configuration-rca:R",  "hb-configuration-rca:U",  "hb-configuration-resources:C",  "hb-configuration-resources:D",  "hb-configuration-resources:R",  "hb-configuration-resources:Rule Editor",  "hb-configuration-resources:U",  "hb-configuration-rules:C",  "hb-configuration-rules:D",  "hb-configuration-rules:R",  "hb-configuration-rules:Rule Editor",  "hb-configuration-rules:U",  "hb-configuration-workflows:C",  "hb-configuration-workflows:D",  "hb-configuration-workflows:R",  "hb-configuration-workflows:U",  "hb-dashboard-alert-severity:R",  "hb-dashboard-alert-status:R",  "hb-dashboard-alert-summary:R",  "hb-dashboard-device-groups:R",  "hb-dashboard-device-status:R",  "hb-dashboard-device-vendors:R",  "hb-dashboard-devicegroup-health:R",  "hb-dashboard-devices:R",  "hb-dashboard-network-health:R",  "hb-dashboard-networkgroups:R",  "hb-dashboard-tsdb:R",  "hb-dashboard-workflows:R",  "hb-deployment-configuration:R",  "hb-deployment-configuration:U",  "hb-dynamic-tagging:C",  "hb-dynamic-tagging:D",  "hb-dynamic-tagging:R",  "hb-dynamic-tagging:U",  "hb-monitor-alarms:R",  "hb-monitor-device-health:R",  "hb-monitor-devicegroup-health:R",  "hb-monitor-graph-grafana:AllAccess",  "hb-monitor-graph:C",  "hb-monitor-graph:D",  "hb-monitor-graph:R",  "hb-monitor-graph:U",  "hb-monitor-network-health:R",  "hb-monitor-reports:R",  "hb-monitor-workflow:C",  "hb-monitor-workflow:D",  "hb-monitor-workflow:R",  "hb-monitor-workflow:U",  "hb-settings-alarm:C",  "hb-settings-alarm:D",  "hb-settings-alarm:R",  "hb-settings-alarm:U",  "hb-settings-datasumm-profiles:C",  "hb-settings-datasumm-profiles:D",  "hb-settings-datasumm-profiles:R",  "hb-settings-datasumm-profiles:U",  "hb-settings-destination:C",  "hb-settings-destination:D",  "hb-settings-destination:R",  "hb-settings-destination:U",  "hb-settings-ingest-byoi:C",  "hb-settings-ingest-byoi:Custom Plugins",  "hb-settings-ingest-byoi:D",  "hb-settings-ingest-byoi:R",  "hb-settings-ingest-byoi:U",  "hb-settings-ingest-frequencyprofile:C",  "hb-settings-ingest-frequencyprofile:D",  "hb-settings-ingest-frequencyprofile:R",  "hb-settings-ingest-frequencyprofile:U",  "hb-settings-ingest-iagent-outbound-ssh:C",  "hb-settings-ingest-iagent-outbound-ssh:D",  "hb-settings-ingest-iagent-outbound-ssh:R",  "hb-settings-ingest-iagent-outbound-ssh:U",  "hb-settings-ingest-ifa:C",  "hb-settings-ingest-ifa:D",  "hb-settings-ingest-ifa:R",  "hb-settings-ingest-ifa:U",  "hb-settings-ingest-native-gpb:R",  "hb-settings-ingest-native-gpb:U",  "hb-settings-ingest-netflow:C",  "hb-settings-ingest-netflow:D",  "hb-settings-ingest-netflow:R",  "hb-settings-ingest-netflow:U",  "hb-settings-ingest-paa:C",  "hb-settings-ingest-paa:D",  "hb-settings-ingest-paa:R",  "hb-settings-ingest-paa:U",  "hb-settings-ingest-sflow:C",  "hb-settings-ingest-sflow:D",  "hb-settings-ingest-sflow:R",  "hb-settings-ingest-sflow:U",  "hb-settings-ingest-snmp-notification:C",  "hb-settings-ingest-snmp-notification:D",  "hb-settings-ingest-snmp-notification:R",  "hb-settings-ingest-snmp-notification:U",  "hb-settings-ingest-syslog:C",  "hb-settings-ingest-syslog:D",  "hb-settings-ingest-syslog:R",  "hb-settings-ingest-syslog:U",  "hb-settings-ingest-tagging-profile:C",  "hb-settings-ingest-tagging-profile:D",  "hb-settings-ingest-tagging-profile:R",  "hb-settings-ingest-tagging-profile:U",  "hb-settings-license:C",  "hb-settings-license:D",  "hb-settings-license:R",  "hb-settings-license:U",  "hb-settings-notification:C",  "hb-settings-notification:D",  "hb-settings-notification:R",  "hb-settings-notification:U",  "hb-settings-report:C",  "hb-settings-report:D",  "hb-settings-report:R",  "hb-settings-report:U",  "hb-settings-retentionpolicy:C",  "hb-settings-retentionpolicy:D",  "hb-settings-retentionpolicy:R",  "hb-settings-retentionpolicy:U",  "hb-settings-rollup-summarization-profile:C",  "hb-settings-rollup-summarization-profile:D",  "hb-settings-rollup-summarization-profile:R",  "hb-settings-rollup-summarization-profile:U",  "hb-settings-scheduler:C",  "hb-settings-scheduler:D",  "hb-settings-scheduler:R",  "hb-settings-scheduler:U",  "hb-settings-security:C",  "hb-settings-security:D",  "hb-settings-security:R",  "hb-settings-trigger\_action:C",  "hb-settings-trigger\_action:D",  "hb-settings-trigger\_action:R",  "hb-settings-trigger\_action:U",  "hb-settings-tsdb:C",  "hb-settings-tsdb:D",  "hb-settings-tsdb:R",  "hb-settings-tsdb:U",  "ns-demand:C",  "ns-demand:D",  "ns-demand:R",  "ns-demand:U",  "ns-maintenance:C",  "ns-maintenance:D",  "ns-maintenance:R",  "ns-maintenance:Simulate",  "ns-maintenance:U",  "ns-pathoptimization:C",  "ns-pathoptimization:R",  "ns-pathoptimization:U",  "ns-planner-desktop:R",  "ns-reports-advanced:R",  "ns-segmentrouting:C",  "ns-segmentrouting:D",  "ns-segmentrouting:R",  "ns-segmentrouting:U",  "ns-service:C",  "ns-service:D",  "ns-service:R",  "ns-service:U",  "ns-settings:C",  "ns-settings:D",  "ns-settings:R",  "ns-settings:Reset",  "ns-settings:Sync",  "ns-settings:U",  "ns-taskscheduler:C",  "ns-taskscheduler:D",  "ns-taskscheduler:R",  "ns-taskscheduler:U",  "ns-topology:C",  "ns-topology:D",  "ns-topology:Diagnostics",  "ns-topology:R",  "ns-topology:U",  "ns-tunnel:Approve",  "ns-tunnel:Autoapprove",  "ns-tunnel:C",  "ns-tunnel:D",  "ns-tunnel:Deploy",  "ns-tunnel:R",  "ns-tunnel:U"  ],  "scope": {  "id": "eecda7d9-d021-4665-9fe9-05348437c091",  "name": "default",  "type": "SP",  "display\_name": "default"  },  "capabilities": [  {  "service": "alarmmanager.MetaRPCService",  "methods": [  "CreateMeta",  "DeleteMeta",  "GetMeta",  "ListMetas",  "UpdateMeta"  ]  },  {  "service": "devicemodelconnector.v1.ConnectorService",  "methods": [  "Publish",  "Reconcile"  ]  },  {  "service": "auditlog.AuditlogService",  "methods": [  "ExportLog",  "GetLog",  "ListLog",  "Subscribe"  ]  },  {  "service": "healthbot.rca-enrichment",  "methods": [  "/api/#/rca/resource-alarms:DELETE",  "/api/#/rca/resource-alarms:GET",  "/api/#/rca/resource-alarms:POST",  "/api/#/rca/resource-alarms:PUT",  "/api/#/rca/resource-dependencies:DELETE",  "/api/#/rca/resource-dependencies:GET",  "/api/#/rca/resource-dependencies:POST",  "/api/#/rca/resource-dependencies:PUT",  "/api/#/rca/resource-recommendations:GET",  "/api/#/rca/resource-recommendations:POST",  "/api/#/rca/resources:DELETE",  "/api/#/rca/resources:GET",  "/api/#/rca/resources:POST",  "/api/#/rca/resources:PUT"  ]  },  {  "service": "alarmmanager.AlarmmanagerService",  "methods": [  "Subscribe"  ]  },  {  "service": "healthbot.mgd",  "methods": [  "/mgd:POST",  "/sshcli:POST"  ]  },  {  "service": "dcs.v1.EdgeService",  "methods": [  "Connect",  "GetConnectionInfo",  "NotifyDeviceConnectionStatus",  "StreamDeviceMessages"  ]  },  {  "service": "devicemodel.DevicemodelService",  "methods": [  "BulkExtRefUpdate",  "BulkListDeviceView",  "CreateConnector",  "CreateDevice",  "CreateDeviceFamily",  "CreateDeviceOperationalState",  "DeleteConnector",  "DeleteDevice",  "DeleteDeviceFamily",  "DeleteDeviceOperationalState",  "ExportComponent",  "ExportDeviceView",  "ExportInterface",  "ExportLicense",  "ExportLicenseFeature",  "ExtRefUpdate",  "GetCaCertificate",  "GetCertificate",  "GetComponent",  "GetConnector",  "GetDevice",  "GetDeviceFamily",  "GetDeviceOperationalState",  "GetDeviceView",  "GetInterface",  "GetLicense",  "GetLicenseFeature",  "GetSubinterface",  "ListCaCertificate",  "ListCertificate",  "ListComponent",  "ListConnector",  "ListDevice",  "ListDeviceFamily",  "ListDeviceOperationalState",  "ListDeviceView",  "ListInterface",  "ListLicense",  "ListLicenseFeature",  "ListSubinterface",  "Subscribe",  "TagDevice",  "UntagDevice",  "UpdateConnector",  "UpdateDevice",  "UpdateDeviceFamily",  "UpdateDeviceOperationalState"  ]  },  {  "service": "devicemodel.DataAccessorExtention",  "methods": [  "GetDeviceCounts",  "GetDeviceCountsByAppExtention"  ]  },  {  "service": "devicemodel.ExportInventory",  "methods": [  "ExportInventory"  ]  },  {  "service": "jobstore.JobstoreManager",  "methods": [  "Purge"  ]  },  {  "service": "dmonproxy.MetricService",  "methods": [  "GetMetrics"  ]  },  {  "service": "iam.IamService",  "methods": [  "CreateEmailTemplate",  "CreateRole",  "DeleteEmailTemplate",  "DeleteOidcSetting",  "DeleteRole",  "DeleteUser",  "DeleteUserGroup",  "GetEmailTemplate",  "GetLdapServer",  "GetOidcSetting",  "GetRole",  "GetSMTPServer",  "GetSaml",  "GetScope",  "GetScopeGroup",  "GetUserGroup",  "GetUserView",  "ListEmailTemplate",  "ListIdentityProvider",  "ListLdapServer",  "ListOidcSetting",  "ListRole",  "ListRoleassignment",  "ListSMTPServer",  "ListSaml",  "ListScope",  "ListScopeGroup",  "ListUserGroup",  "ListUserView",  "Subscribe",  "UpdateEmailTemplate",  "UpdateRole"  ]  },  {  "service": "iam.IamRPCService",  "methods": [  "CheckLDAPAuthentication",  "CheckLDAPNetworkConnection",  "CheckOIDCServerConnection",  "CheckSAMLServerConnection",  "ConfigureLDAPServer",  "ConfigureSMTPServer",  "CreateOpenIDCSetting",  "CreateSAMLSetting",  "CreateScope",  "CreateUser",  "CreateUserGroup",  "DeleteOpenIDCSetting",  "DeleteSAMLSetting",  "DeleteScope",  "DeleteUser",  "EditOpenIDCSetting",  "EditSAMLSetting",  "EditUser",  "EditUserGroup",  "Export",  "GetCapabilityCatalog",  "GetOidcRedirectAuthUri",  "GetPortalURL",  "GetUserDetailsData",  "GetUsersByRoleAssignmentAndStatus",  "Import",  "SearchIdentityProviderGroups",  "SendActivateUserEmail",  "SendTestMail",  "UpdateAuthenticationSettings",  "UpdateIdentityProviderPriority",  "UpdateScope"  ]  },  {  "service": "healthbot.alerta",  "methods": [  "/alertaapi/alert/#/action:PUT",  "/alertaapi/alert/#/attributes:PUT",  "/alertaapi/alert/#/status:PUT",  "/alertaapi/alert/#/tag:PUT",  "/alertaapi/alert/#/untag:PUT",  "/alertaapi/alert/#:DELETE",  "/alertaapi/alert/#:GET",  "/alertaapi/alert:POST",  "/alertaapi/alerts/count:GET",  "/alertaapi/alerts/history:GET",  "/alertaapi/alerts/top10/count:GET",  "/alertaapi/alerts/top10/flapping:GET",  "/alertaapi/alerts/top10/standing:GET",  "/alertaapi/alerts:DELETE",  "/alertaapi/alerts:GET",  "/alertaapi/blackout/#:DELETE",  "/alertaapi/blackout:POST",  "/alertaapi/blackouts:GET"  ]  },  {  "service": "gnmi.gNMI",  "methods": [  "\*"  ]  },  {  "service": "healthbot.reports",  "methods": [  "/api/#/reports/\*:GET"  ]  },  {  "service": "healthbot.api-server",  "methods": [  "/api/#(/config)?/data-store/grafana:GET",  "/api/#/data/\*:GET",  "/api/#/debug/\*:GET",  "/api/#/debug/configuration:POST",  "/api/#/debug/jobs:GET",  "/api/#/debug/scenario/#:POST",  "/api/#/debug/scenario/device-reachability:POST",  "/api/#/debug/scenario/ingest-test:POST",  "/api/#/deployed-device-details:POST",  "/api/#/device-group/#/status:GET",  "/api/#/device-group/#/trigger\_info:GET",  "/api/#/device/#/trigger\_info:GET",  "/api/#/event/#:GET",  "/api/#/event/device-group/\*:GET",  "/api/#/event/network-group/\*:GET",  "/api/#/event:GET",  "/api/#/events:GET",  "/api/#/grafana/backup/\*:GET",  "/api/#/grafana/login/\*:GET",  "/api/#/grafana/restore/\*:POST",  "/api/#/health-tree/#:GET",  "/api/#/health-tree/device-group/#:GET",  "/api/#/health-tree/network-group/#:GET",  "/api/#/health:GET",  "/api/#/license/key/#/contents:GET",  "/api/#/license/key/#:DELETE",  "/api/#/license/key/#:GET",  "/api/#/license/keys/contents:GET",  "/api/#/license/keys:DELETE",  "/api/#/license/keys:GET",  "/api/#/license/keys:POST",  "/api/#/license/keys:PUT",  "/api/#/license/status:GET",  "/api/#/logs/\*:GET",  "/api/#/network-group/#/status:GET",  "/api/#/network-group/#/trigger\_info:GET",  "/api/#/nodes:GET",  "/api/#/orchestrator:GET",  "/api/#/sensor(/config)?/device-group/#:GET",  "/api/#/tsdb-counters:GET",  "/api/#/tsdb/query:GET",  "/api/#/tsdb/query:POST",  "/api/#/workflow-instance/#:DELETE",  "/api/#/workflow-instance/#:GET",  "/api/#/workflow-instance/#:POST",  "/api/#/workflow-instance/#:PUT",  "/api/#/workflow-instances:DELETE",  "/api/#/workflow-instances:GET",  "/api/#/workflow-instances:PUT",  "/data/database/table/\*:GET",  "/field-capture:GET",  "/inspect/command-rpc/table:POST"  ]  },  {  "service": "dpm.ImageManager",  "methods": [  "AddImage",  "DeleteCertificate",  "DeleteImages",  "DeployCACertificate",  "DeployImage",  "DeployLicense",  "DeployLocalCertificate",  "EnableDefaultCAProfile",  "StageImage",  "UploadImageFile",  "UploadImageURL"  ]  },  {  "service": "auditlog.AuditlogRPCService",  "methods": [  "CreateLogEntry",  "Purge",  "PurgePolicy"  ]  },  {  "service": "devicemanager.ConfigTemplateService",  "methods": [  "AssignToDeviceTemplate",  "ConfigureInitValues",  "CreateWithViewdef",  "Delete",  "Deploy",  "DeployAssociations",  "DeployAsync",  "EditWithViewdef",  "GenerateSchema",  "GetAssociations",  "ListConfigTemplateView",  "Preview",  "SaveAssociations",  "UnassignFromDeviceTemplate",  "Validate",  "ValidateAssociations",  "ValidateAsync"  ]  },  {  "service": "devicemanager.ConfigOperationService",  "methods": [  "\*",  "GetDeviceArchivedConfigVersion",  "GetDeviceConfiguration",  "GetEffectiveState"  ]  },  {  "service": "devicemanager.DevicemanagerService",  "methods": [  "BulkListConfigTemplate",  "CreateConfigTemplate",  "CreateConfigTemplateDeployment",  "CreateConfigTemplateViewdef",  "CreateDeviceTemplate",  "DeleteActivationIPInfo",  "DeleteConfigArchive",  "DeleteConfigResource",  "DeleteConfigTemplateDeployment",  "DeleteConfiguration",  "DeleteDeviceActivationInfo",  "DeleteDeviceTemplate",  "GetCaCertificate",  "GetCertificate",  "GetComponent",  "GetConfigArchive",  "GetConfigResource",  "GetConfigTemplate",  "GetConfigTemplateDeployment",  "GetConfigTemplateViewdef",  "GetConfiguration",  "GetDevice",  "GetDeviceTemplate",  "GetInterface",  "GetLicense",  "GetLicenseFeature",  "GetSubinterface",  "ListActivationIPInfo",  "ListCaCertificate",  "ListCertificate",  "ListComponent",  "ListConfigArchive",  "ListConfigResource",  "ListConfigResourceRef",  "ListConfigTemplate",  "ListConfigTemplateDeployment",  "ListConfigTemplateViewdef",  "ListConfiguration",  "ListDevice",  "ListDeviceActivationInfo",  "ListDeviceFamily",  "ListDeviceTemplate",  "ListInterface",  "ListLicense",  "ListLicenseFeature",  "ListSubinterface",  "Subscribe",  "UpdateActivationIPInfo",  "UpdateConfigTemplate",  "UpdateConfigTemplateDeployment",  "UpdateConfigTemplateViewdef",  "UpdateDeviceActivationInfo",  "UpdateDeviceTemplate",  "ValidateConfig"  ]  },  {  "service": "telemetrymanager.TelemetryManagerService",  "methods": [  "Debug",  "GetConfig",  "GetSchema",  "InitConfig",  "Query",  "RemoveConfig",  "Subscribe",  "UpdateConfig"  ]  },  {  "service": "healthbot.grafana",  "methods": [  "/grafana/\*:DELETE",  "/grafana/\*:GET",  "/grafana/\*:POST",  "/grafana/\*:PUT"  ]  },  {  "service": "alarmmanager.AlarmRPCService",  "methods": [  "ClearAlarm",  "GetAlarm",  "GetAlarms",  "ListAlarms",  "PurgeHistoricalAlarms",  "QueryAlarms",  "QueryHistoricalAlarms",  "RaiseAlarm",  "SummaryHistoricalAlarms",  "SyncAlarms",  "UpdateAlarm"  ]  },  {  "service": "jobmanager.JobService",  "methods": [  "CancelJob",  "CreateJob",  "CreateJobSchedule"  ]  },  {  "service": "devicemanager.InventoryService",  "methods": [  "DeleteArchivedConfigVersion",  "DiffArchivedConfigVersions",  "GetArchivedConfigVersion",  "GetDeviceInfo",  "ListDeviceConfiguration",  "Subscribe",  "SyncInventory",  "SyncInventoryAsync",  "UpdateActivationIPInfo",  "UpdateArchivedConfigVersion",  "UpdateDeviceActivationInfo"  ]  },  {  "service": "devicemanager.DeviceLCMService",  "methods": [  "DeleteDevicesAsync",  "DeleteSingleDevice"  ]  },  {  "service": "alarmmanager.NotificationRPCService",  "methods": [  "CreateSubscription",  "DeleteSubscription",  "GetSubscription",  "ListSubscriptions",  "UpdateSubscription"  ]  },  {  "service": "jobstore.JobstoreService",  "methods": [  "BulkListJob",  "CreateJobPurgePolicy",  "DeleteJobPurgePolicy",  "GetJob",  "GetJobPurgePolicy",  "GetTask",  "ListJob",  "ListJobPurgePolicy",  "ListTask",  "Subscribe",  "UpdateJobPurgePolicy"  ]  },  {  "service": "devicemanager.ConfigResourceService",  "methods": [  "\*"  ]  },  {  "service": "devicemanager.DeviceOnboardingService",  "methods": [  "ActivateReachableDevices",  "AddDevices",  "DiscoverDevices",  "GetBootstrapConfig",  "GetSupportedVendors"  ]  },  {  "service": "dpmEdge.v1.DPMEdgeService",  "methods": [  "Connect"  ]  },  {  "service": "jobmanager",  "methods": [  "activate-workflow-definition",  "cancel-job",  "create-job",  "create-job-schedule",  "create-workflow-definition",  "delete-workflow-definition",  "purge-job",  "retry-job",  "update-workflow-definition",  "workflow-definition",  "workflow-version"  ]  },  {  "service": "healthbot.config-server",  "methods": [  "/api/#(/config)?/configuration/check/device-group/#:POST",  "/api/#(/config)?/configuration/check/network-group/#:POST",  "/api/#(/config)?/configuration/jobs:GET",  "/api/#(/config)?/configuration:DELETE",  "/api/#(/config)?/configuration:GET",  "/api/#(/config)?/configuration:POST",  "/api/#(/config)?/data-store/#:DELETE",  "/api/#(/config)?/data-store/#:GET",  "/api/#(/config)?/data-store/#:POST",  "/api/#(/config)?/data-store/#:PUT",  "/api/#(/config)?/data-store/dashboard-config:GET",  "/api/#(/config)?/data-store/dashboard-config:POST",  "/api/#(/config)?/data-store/dashboard-config:PUT",  "/api/#(/config)?/data-store/grafana:DELETE",  "/api/#(/config)?/data-store/grafana:POST",  "/api/#(/config)?/deployment:GET",  "/api/#(/config)?/deployment:PUT",  "/api/#(/config)?/device-group/#/device:DELETE",  "/api/#(/config)?/device-group/#/facts:GET",  "/api/#(/config)?/device-group/#/status:GET",  "/api/#(/config)?/device-group/#:DELETE",  "/api/#(/config)?/device-group/#:GET",  "/api/#(/config)?/device-group/#:POST",  "/api/#(/config)?/device-group/#:PUT",  "/api/#(/config)?/device-group:GET",  "/api/#(/config)?/device-groups:DELETE",  "/api/#(/config)?/device-groups:GET",  "/api/#(/config)?/device-groups:POST",  "/api/#(/config)?/device/#/facts:GET",  "/api/#(/config)?/device/#:GET",  "/api/#(/config)?/device/#:POST",  "/api/#(/config)?/device/#:PUT",  "/api/#(/config)?/device:GET",  "/api/#(/config)?/device:POST",  "/api/#(/config)?/devices/facts:GET",  "/api/#(/config)?/devices:GET",  "/api/#(/config)?/devices:POST",  "/api/#(/config)?/dynamic-tagging/key:DELETE",  "/api/#(/config)?/dynamic-tagging/key:GET",  "/api/#(/config)?/dynamic-tagging/key:POST",  "/api/#(/config)?/dynamic-tagging/key:PUT",  "/api/#(/config)?/dynamic-tagging/keys:DELETE",  "/api/#(/config)?/dynamic-tagging/keys:GET",  "/api/#(/config)?/dynamic-tagging/keys:POST",  "/api/#(/config)?/dynamic-tagging/keys:PUT",  "/api/#(/config)?/files/certificates/\*:DELETE",  "/api/#(/config)?/files/certificates/\*:POST",  "/api/#(/config)?/files/helper-files/#:DELETE",  "/api/#(/config)?/files/helper-files/#:GET",  "/api/#(/config)?/files/helper-files/#:POST",  "/api/#(/config)?/files/helper-files/backup:GET",  "/api/#(/config)?/files/helper-files:GET",  "/api/#(/config)?/ingest-settings/byoi/\*:DELETE",  "/api/#(/config)?/ingest-settings/byoi/\*:GET",  "/api/#(/config)?/ingest-settings/byoi/\*:POST",  "/api/#(/config)?/ingest-settings/byoi/\*:PUT",  "/api/#(/config)?/ingest-settings/byoi/ingest-mappings:DELETE",  "/api/#(/config)?/ingest-settings/data-enrichment/tagging-profile/\*:DELETE",  "/api/#(/config)?/ingest-settings/data-enrichment/tagging-profile/\*:GET",  "/api/#(/config)?/ingest-settings/data-enrichment/tagging-profile/\*:POST",  "/api/#(/config)?/ingest-settings/data-enrichment/tagging-profile/\*:PUT",  "/api/#(/config)?/ingest-settings/data-enrichment/tagging-profiles:GET",  "/api/#(/config)?/ingest-settings/flow/\*:DELETE",  "/api/#(/config)?/ingest-settings/flow/\*:GET",  "/api/#(/config)?/ingest-settings/flow/\*:POST",  "/api/#(/config)?/ingest-settings/flow/\*:PUT",  "/api/#(/config)?/ingest-settings/frequency-profile/\*:DELETE",  "/api/#(/config)?/ingest-settings/frequency-profile/\*:GET",  "/api/#(/config)?/ingest-settings/frequency-profile/\*:POST",  "/api/#(/config)?/ingest-settings/frequency-profile/\*:PUT",  "/api/#(/config)?/ingest-settings/frequency-profiles:GET",  "/api/#(/config)?/ingest-settings/syslog/\*:DELETE",  "/api/#(/config)?/ingest-settings/syslog/\*:GET",  "/api/#(/config)?/ingest-settings/syslog/\*:POST",  "/api/#(/config)?/ingest-settings/syslog/\*:PUT",  "/api/#(/config)?/ingest-settings:DELETE",  "/api/#(/config)?/ingest-settings:GET",  "/api/#(/config)?/ingest-settings:POST",  "/api/#(/config)?/ingest-settings:PUT",  "/api/#(/config)?/ingest/byoi/\*:DELETE",  "/api/#(/config)?/ingest/byoi/\*:GET",  "/api/#(/config)?/ingest/byoi/\*:POST",  "/api/#(/config)?/ingest/byoi/\*:PUT",  "/api/#(/config)?/ingest/data-enrichment/tagging-profile/\*:DELETE",  "/api/#(/config)?/ingest/data-enrichment/tagging-profile/\*:GET",  "/api/#(/config)?/ingest/data-enrichment/tagging-profile/\*:POST",  "/api/#(/config)?/ingest/data-enrichment/tagging-profile/\*:PUT",  "/api/#(/config)?/ingest/data-enrichment/tagging-profiles:DELETE",  "/api/#(/config)?/ingest/data-enrichment/tagging-profiles:GET",  "/api/#(/config)?/ingest/data-enrichment/tagging-profiles:POST",  "/api/#(/config)?/ingest/data-enrichment/tagging-profiles:PUT",  "/api/#(/config)?/ingest/flow/\*:DELETE",  "/api/#(/config)?/ingest/flow/\*:GET",  "/api/#(/config)?/ingest/flow/\*:POST",  "/api/#(/config)?/ingest/flow/\*:PUT",  "/api/#(/config)?/ingest/flow:GET",  "/api/#(/config)?/ingest/frequency-profile/\*:DELETE",  "/api/#(/config)?/ingest/frequency-profile/\*:GET",  "/api/#(/config)?/ingest/frequency-profile/\*:POST",  "/api/#(/config)?/ingest/frequency-profile/\*:PUT",  "/api/#(/config)?/ingest/frequency-profiles:GET",  "/api/#(/config)?/ingest/ifa/\*:DELETE",  "/api/#(/config)?/ingest/ifa/\*:GET",  "/api/#(/config)?/ingest/ifa/\*:Get",  "/api/#(/config)?/ingest/ifa/\*:POST",  "/api/#(/config)?/ingest/ifa/\*:PUT",  "/api/#(/config)?/ingest/native-gpb:GET",  "/api/#(/config)?/ingest/native-gpb:PUT",  "/api/#(/config)?/ingest/outbound-ssh:DELETE",  "/api/#(/config)?/ingest/outbound-ssh:GET",  "/api/#(/config)?/ingest/outbound-ssh:POST",  "/api/#(/config)?/ingest/outbound-ssh:PUT",  "/api/#(/config)?/ingest/paa\*:GET",  "/api/#(/config)?/ingest/paa/\*:DELETE",  "/api/#(/config)?/ingest/paa/\*:GET",  "/api/#(/config)?/ingest/paa/\*:POST",  "/api/#(/config)?/ingest/paa/\*:PUT",  "/api/#(/config)?/ingest/paa:GET",  "/api/#(/config)?/ingest/paa:Get",  "/api/#(/config)?/ingest/sflow/\*:DELETE",  "/api/#(/config)?/ingest/sflow/\*:GET",  "/api/#(/config)?/ingest/sflow/\*:POST",  "/api/#(/config)?/ingest/sflow/\*:PUT",  "/api/#(/config)?/ingest/sflow:GET",  "/api/#(/config)?/ingest/snmp-notification/\*:DELETE",  "/api/#(/config)?/ingest/snmp-notification/\*:GET",  "/api/#(/config)?/ingest/snmp-notification/\*:POST",  "/api/#(/config)?/ingest/snmp-notification/\*:PUT",  "/api/#(/config)?/ingest/syslog/\*:DELETE",  "/api/#(/config)?/ingest/syslog/\*:GET",  "/api/#(/config)?/ingest/syslog/\*:Get",  "/api/#(/config)?/ingest/syslog/\*:POST",  "/api/#(/config)?/ingest/syslog/\*:PUT",  "/api/#(/config)?/ingest:DELETE",  "/api/#(/config)?/ingest:GET",  "/api/#(/config)?/ingest:POST",  "/api/#(/config)?/ingest:PUT",  "/api/#(/config)?/initialize:POST",  "/api/#(/config)?/instances-schedule-state/\*:GET",  "/api/#(/config)?/instances-schedule-state/\*:PUT",  "/api/#(/config)?/network-group/#/status:GET",  "/api/#(/config)?/network-group/#/variable:DELETE",  "/api/#(/config)?/network-group/#:DELETE",  "/api/#(/config)?/network-group/#:GET",  "/api/#(/config)?/network-group/#:POST",  "/api/#(/config)?/network-group/#:PUT",  "/api/#(/config)?/network-group:DELETE",  "/api/#(/config)?/network-group:GET",  "/api/#(/config)?/network-group:PUT",  "/api/#(/config)?/network-groups/#:GET",  "/api/#(/config)?/network-groups:DELETE",  "/api/#(/config)?/network-groups:GET",  "/api/#(/config)?/network-groups:POST",  "/api/#(/config)?/notification/#:DELETE",  "/api/#(/config)?/notification/#:GET",  "/api/#(/config)?/notification/#:POST",  "/api/#(/config)?/notification/#:PUT",  "/api/#(/config)?/notification:GET",  "/api/#(/config)?/notifications:DELETE",  "/api/#(/config)?/notifications:GET",  "/api/#(/config)?/notifications:POST",  "/api/#(/config)?/notifications:PUT",  "/api/#(/config)?/playbook/#:DELETE",  "/api/#(/config)?/playbook/#:GET",  "/api/#(/config)?/playbook/#:POST",  "/api/#(/config)?/playbook/#:PUT",  "/api/#(/config)?/playbook:GET",  "/api/#(/config)?/playbooks:DELETE",  "/api/#(/config)?/playbooks:GET",  "/api/#(/config)?/playbooks:POST",  "/api/#(/config)?/playbooks:PUT",  "/api/#(/config)?/profile/data-summarization/raw/#:DELETE",  "/api/#(/config)?/profile/data-summarization/raw/#:POST",  "/api/#(/config)?/profile/data-summarization/raw/#:PUT",  "/api/#(/config)?/profile/data-summarizations/raw:GET",  "/api/#(/config)?/profile/rollup-summarization/field-profile/#:DELETE",  "/api/#(/config)?/profile/rollup-summarization/field-profile/#:GET",  "/api/#(/config)?/profile/rollup-summarization/field-profile/#:POST",  "/api/#(/config)?/profile/rollup-summarization/field-profile/#:PUT",  "/api/#(/config)?/profile/rollup-summarization/field-profile:GET",  "/api/#(/config)?/profile/security/\*:GET",  "/api/#(/config)?/profile/security/ca-profile/#:DELETE",  "/api/#(/config)?/profile/security/ca-profile/#:POST",  "/api/#(/config)?/profile/security/ca-profiles:GET",  "/api/#(/config)?/profile/security/local-certificate/#:DELETE",  "/api/#(/config)?/profile/security/local-certificate/#:POST",  "/api/#(/config)?/profile/security/local-certificates:GET",  "/api/#(/config)?/profile/security/ssh-key-profile/#:DELETE",  "/api/#(/config)?/profile/security/ssh-key-profile/#:POST",  "/api/#(/config)?/profile/security/ssh-key-profiles:GET",  "/api/#(/config)?/profiles:DELETE",  "/api/#(/config)?/profiles:GET",  "/api/#(/config)?/profiles:POST",  "/api/#(/config)?/retention-policies:DELETE",  "/api/#(/config)?/retention-policies:GET",  "/api/#(/config)?/retention-policies:POST",  "/api/#(/config)?/retention-policies:PUT",  "/api/#(/config)?/retention-policy/#:DELETE",  "/api/#(/config)?/retention-policy/#:POST",  "/api/#(/config)?/retention-policy/#:PUT",  "/api/#(/config)?/retention-policy/\*:GET",  "/api/#(/config)?/sensor/\*:GET",  "/api/#(/config)?/sensors:GET",  "/api/#(/config)?/services/device-group/#:DELETE",  "/api/#(/config)?/services/device-group/#:POST",  "/api/#(/config)?/services/device-group:GET",  "/api/#(/config)?/services/network-group/#:DELETE",  "/api/#(/config)?/services/network-group/#:POST",  "/api/#(/config)?/services/network-group:GET",  "/api/#(/config)?/system-settings/report-generation/\*:DELETE",  "/api/#(/config)?/system-settings/report-generation/\*:GET",  "/api/#(/config)?/system-settings/report-generation/\*:POST",  "/api/#(/config)?/system-settings/report-generation/\*:PUT",  "/api/#(/config)?/system-settings/report-generation/destinations:DELETE",  "/api/#(/config)?/system-settings/report-generation/destinations:POST",  "/api/#(/config)?/system-settings/report-generation/reports:DELETE",  "/api/#(/config)?/system-settings/report-generation/reports:GET",  "/api/#(/config)?/system-settings/report-generation/reports:POST",  "/api/#(/config)?/system-settings/scheduler/#:DELETE",  "/api/#(/config)?/system-settings/scheduler/#:GET",  "/api/#(/config)?/system-settings/scheduler/#:POST",  "/api/#(/config)?/system-settings/scheduler/#:PUT",  "/api/#(/config)?/system-settings/schedulers:DELETE",  "/api/#(/config)?/system-settings/schedulers:GET",  "/api/#(/config)?/system-settings/schedulers:POST",  "/api/#(/config)?/system-settings/schedulers:PUT",  "/api/#(/config)?/system-settings:DELETE",  "/api/#(/config)?/system-settings:GET",  "/api/#(/config)?/system-settings:POST",  "/api/#(/config)?/system-settings:PUT",  "/api/#(/config)?/system/report-generation/destination/#:DELETE",  "/api/#(/config)?/system/report-generation/destination/#:GET",  "/api/#(/config)?/system/report-generation/destination/#:POST",  "/api/#(/config)?/system/report-generation/destination/#:PUT",  "/api/#(/config)?/system/report-generation/destinations:DELETE",  "/api/#(/config)?/system/report-generation/destinations:GET",  "/api/#(/config)?/system/report-generation/destinations:POST",  "/api/#(/config)?/system/report-generation/destinations:PUT",  "/api/#(/config)?/system/report-generation/report/#:DELETE",  "/api/#(/config)?/system/report-generation/report/#:GET",  "/api/#(/config)?/system/report-generation/report/#:POST",  "/api/#(/config)?/system/report-generation/report/#:PUT",  "/api/#(/config)?/system/report-generation/reports:DELETE",  "/api/#(/config)?/system/report-generation/reports:GET",  "/api/#(/config)?/system/report-generation/reports:POST",  "/api/#(/config)?/system/report-generation/reports:PUT",  "/api/#(/config)?/system/scheduler/#:DELETE",  "/api/#(/config)?/system/scheduler/#:GET",  "/api/#(/config)?/system/scheduler/#:POST",  "/api/#(/config)?/system/scheduler/#:PUT",  "/api/#(/config)?/system/scheduler:DELETE",  "/api/#(/config)?/system/schedulers:DELETE",  "/api/#(/config)?/system/schedulers:GET",  "/api/#(/config)?/system/schedulers:POST",  "/api/#(/config)?/system/trigger\_action:DELETE",  "/api/#(/config)?/system/trigger\_action:GET",  "/api/#(/config)?/system/trigger\_action:POST",  "/api/#(/config)?/system/trigger\_action:PUT",  "/api/#(/config)?/system/tsdb:DELETE",  "/api/#(/config)?/system/tsdb:GET",  "/api/#(/config)?/system/tsdb:POST",  "/api/#(/config)?/system/tsdb:PUT",  "/api/#(/config)?/system:DELETE",  "/api/#(/config)?/system:GET",  "/api/#(/config)?/system:POST",  "/api/#(/config)?/system:PUT",  "/api/#(/config)?/topic/\*:DELETE",  "/api/#(/config)?/topic/\*:GET",  "/api/#(/config)?/topic/\*:POST",  "/api/#(/config)?/topic/\*:PUT",  "/api/#(/config)?/topic:GET",  "/api/#(/config)?/topics:DELETE",  "/api/#(/config)?/topics:GET",  "/api/#(/config)?/topics:POST",  "/api/#(/config)?/workflow/#:DELETE",  "/api/#(/config)?/workflow/#:GET",  "/api/#(/config)?/workflow/#:POST",  "/api/#(/config)?/workflow/#:PUT",  "/api/#(/config)?/workflow:GET",  "/api/#(/config)?/workflows:DELETE",  "/api/#(/config)?/workflows:GET",  "/api/#(/config)?/workflows:POST",  "/api/#(/config)?/workflows:PUT",  "/api/#(config)?/devices:GET",  "/api/#/health-tree/#:GET",  "/api/#/system-details:GET"  ]  },  {  "service": "northstar.ns-web",  "methods": [  "/NorthstarPlanner/API/v1/tenant/1:DELETE",  "/NorthstarPlanner/API/v1/tenant/1:GET",  "/NorthstarPlanner/API/v1/tenant/1:PATCH",  "/NorthstarPlanner/API/v1/tenant/1:POST",  "/NorthstarPlanner/API/v1/tenant/1:PUT",  "/networks/files:GET",  "/networks/import:POST",  "/networks/importConfig:POST",  "/networks/report/ROOT/L2\_LinkUtilChange.r0:GET",  "/networks/report/ROOT/L2\_PathAnalysis.r0:GET",  "/networks/report/ROOT/L2\_PathChange.r0:GET",  "/networks/report/ROOT/LSP\_status.x:GET",  "/networks/report/ROOT/PCS\_pathOptimization.x:GET",  "/networks/report/current/ConfigIntegrityChecks.x:GET",  "/networks/report/demand:GET",  "/networks/report/inventory:GET",  "/networks/report/planner:GET",  "/planner/activebaselinedetail:GET",  "/planner/activebaselines:GET",  "/planner/hardwaretypemapping:GET",  "/planner/image:GET",  "/planner/image:PUT",  "/planner/network:DELETE",  "/planner/networkdata:DELETE",  "/planner/networkdata:GET",  "/planner/networkdata:PATCH",  "/planner/networkdata:POST",  "/planner/networkdata:PUT",  "/planner/sessiondata:DELTE",  "/planner/sessiondata:GET",  "/planner/sessiondata:PATCH",  "/planner/sessiondata:POST",  "/planner/sessiondata:PUT",  "/planner/spec/copy:POST",  "/planner/spec/delete:DELETE",  "/planner/spec/move:POST",  "/planner/spec/spawn:POST",  "/planner/spec:GET",  "/planner/spec:POST",  "/planner/spec:PUT",  "/restconf/data/ietf-network:networks:GET",  "/restconf/data/ietf-te:te/globals/named-admin-groups/named-admin-group:PATCH",  "/restconf/data/ietf-te:te/globals/named-admin-groups:DELETE",  "/restconf/data/ietf-te:te/globals/named-admin-groups:GET",  "/restconf/data/ietf-te:te/globals/named-admin-groups:POST",  "/restconf/data/ietf-te:te/globals/named-admin-groups:PUT",  "/restconf/data/ietf-te:te:GET",  "/restconf/data/ietf-yang-library:modules-state:GET",  "/restconf/data/juniper-pathfinder-topology-group:topology-groups:DELETE",  "/restconf/data/juniper-pathfinder-topology-group:topology-groups:GET",  "/restconf/data/juniper-pathfinder-topology-group:topology-groups:POST",  "/restconf/data/juniper-pathfinder-topology-group:topology-groups:PUT",  "/restconf/operational/ietf-network:networks:GET",  "/restconf/operational/ietf-te:te:GET",  "/restconf/operations/ietf-te-operations:tunnel-atomic-create:POST",  "/restconf/operations/ietf-te-operations:tunnel-atomic-delete:POST",  "/restconf:GET",  "/traffic-engineering/api/netconf/v2/profiles/netconf/reconnect:POST",  "/traffic-engineering/api/netconf/v2/profiles:GET",  "/traffic-engineering/api/private/collectionRest/v1/content:GET",  "/traffic-engineering/api/private/collectionRest/v1/count:GET",  "/traffic-engineering/api/private/collectionRest/v1/datetime:GET",  "/traffic-engineering/api/private/collectionRest/v1/deleteInstance:DELETE",  "/traffic-engineering/api/private/collectionRest/v1/deleteTask:DELETE",  "/traffic-engineering/api/private/collectionRest/v1/list:GET",  "/traffic-engineering/api/private/collectionRest/v1/taskStatus:GET",  "/traffic-engineering/api/private/config/v1:DELETE",  "/traffic-engineering/api/private/config/v1:GET",  "/traffic-engineering/api/private/config/v1:POST",  "/traffic-engineering/api/private/diagnostic:GET",  "/traffic-engineering/api/private/diagnostic:POST",  "/traffic-engineering/api/private/history/v1:GET",  "/traffic-engineering/api/private/networks/report/demand:GET",  "/traffic-engineering/api/private/planning/pcsfm:DELETE",  "/traffic-engineering/api/private/planning/pcsfm:GET",  "/traffic-engineering/api/private/planning/pcsfm:POST",  "/traffic-engineering/api/private/planning/pcsfm:PUT",  "/traffic-engineering/api/private/planning/pcsmh/get:GET",  "/traffic-engineering/api/private/planning/pcsmh/get\_dparam:GET",  "/traffic-engineering/api/private/planning/pcsmh/send:GET",  "/traffic-engineering/api/private/planning/pcsmh/send:POST",  "/traffic-engineering/api/private/planning/pcsmh/send:PUT",  "/traffic-engineering/api/private/planning/pcsmh/set:GET",  "/traffic-engineering/api/private/planning/pcsmh/set:PUT",  "/traffic-engineering/api/private/planning/pcsmh/set\_dparam:PUT",  "/traffic-engineering/api/private/resetTopo:GET",  "/traffic-engineering/api/private/safemodestatus:GET",  "/traffic-engineering/api/private/specRest/v1/nwlist:GET",  "/traffic-engineering/api/private/specRest/v1/specfile:GET",  "/traffic-engineering/api/private/syncTopo:GET",  "/traffic-engineering/api/private/textViewer:GET",  "/traffic-engineering/api/private/webstart:POST",  "/traffic-engineering/api/private/workorder/v1:AUTOAPPROVE",  "/traffic-engineering/api/private/workorder/v1:GET",  "/traffic-engineering/api/private/workorder/v1:POST",  "/traffic-engineering/api/scheduler/v2/addtask:POST",  "/traffic-engineering/api/scheduler/v2/deletetask:POST",  "/traffic-engineering/api/scheduler/v2/removeaddtask:POST",  "/traffic-engineering/api/scheduler/v2/tasklist:GET",  "/traffic-engineering/api/scheduler/v2/tasks:GET",  "/traffic-engineering/api/scheduler/v2/taskshistory:GET",  "/traffic-engineering/api/scheduler/v2/taskstatus:GET",  "/traffic-engineering/api/scheduler/v2/updatetask:POST",  "/traffic-engineering/api/statistics/v2/as\_demands:POST",  "/traffic-engineering/api/statistics/v2/demands:POST",  "/traffic-engineering/api/statistics/v2/device:GET",  "/traffic-engineering/api/statistics/v2/device:POST",  "/traffic-engineering/api/statistics/v2/interfaces:GET",  "/traffic-engineering/api/statistics/v2/interfaces:POST",  "/traffic-engineering/api/statistics/v2/ipe-demands:POST",  "/traffic-engineering/api/statistics/v2/netflow/vpn\_demands:POST",  "/traffic-engineering/api/statistics/v2/prefixes:POST",  "/traffic-engineering/api/statistics/v2/sid:POST",  "/traffic-engineering/api/statistics/v2/te-lsps:GET",  "/traffic-engineering/api/statistics/v2/te-lsps:POST",  "/traffic-engineering/api/topology-filter/v1:DELETE",  "/traffic-engineering/api/topology-filter/v1:GET",  "/traffic-engineering/api/topology-filter/v1:POST",  "/traffic-engineering/api/topology-filter/v1:PUT",  "/traffic-engineering/api/topology/v2/\*anycastGroups:DELETE",  "/traffic-engineering/api/topology/v2/\*anycastGroups:GET",  "/traffic-engineering/api/topology/v2/\*anycastGroups:POST",  "/traffic-engineering/api/topology/v2/\*anycastGroups:PUT",  "/traffic-engineering/api/topology/v2/\*demands:DELETE",  "/traffic-engineering/api/topology/v2/\*demands:GET",  "/traffic-engineering/api/topology/v2/\*demands:PATCH",  "/traffic-engineering/api/topology/v2/\*demands:POST",  "/traffic-engineering/api/topology/v2/\*demands:PUT",  "/traffic-engineering/api/topology/v2/\*facilities:DELETE",  "/traffic-engineering/api/topology/v2/\*facilities:GET",  "/traffic-engineering/api/topology/v2/\*facilities:POST",  "/traffic-engineering/api/topology/v2/\*facilities:PUT",  "/traffic-engineering/api/topology/v2/\*interfaces:GET",  "/traffic-engineering/api/topology/v2/\*interfaces:POST",  "/traffic-engineering/api/topology/v2/\*links:DELETE",  "/traffic-engineering/api/topology/v2/\*links:GET",  "/traffic-engineering/api/topology/v2/\*links:PATCH",  "/traffic-engineering/api/topology/v2/\*links:POST",  "/traffic-engineering/api/topology/v2/\*links:PUT",  "/traffic-engineering/api/topology/v2/\*maintenances:DELETE",  "/traffic-engineering/api/topology/v2/\*maintenances:GET",  "/traffic-engineering/api/topology/v2/\*maintenances:PATCH",  "/traffic-engineering/api/topology/v2/\*maintenances:POST",  "/traffic-engineering/api/topology/v2/\*maintenances:PUT",  "/traffic-engineering/api/topology/v2/\*nodes:DELETE",  "/traffic-engineering/api/topology/v2/\*nodes:GET",  "/traffic-engineering/api/topology/v2/\*nodes:PATCH",  "/traffic-engineering/api/topology/v2/\*nodes:POST",  "/traffic-engineering/api/topology/v2/\*nodes:PUT",  "/traffic-engineering/api/topology/v2/\*p2mp:DELETE",  "/traffic-engineering/api/topology/v2/\*p2mp:GET",  "/traffic-engineering/api/topology/v2/\*p2mp:PATCH",  "/traffic-engineering/api/topology/v2/\*p2mp:POST",  "/traffic-engineering/api/topology/v2/\*p2mp:PUT",  "/traffic-engineering/api/topology/v2/\*pathComputation:POST",  "/traffic-engineering/api/topology/v2/\*rpc/diverseTreeDesign:POST",  "/traffic-engineering/api/topology/v2/\*rpc/optimize:POST",  "/traffic-engineering/api/topology/v2/\*rpc/simulation:GET",  "/traffic-engineering/api/topology/v2/\*rpc/simulation:POST",  "/traffic-engineering/api/topology/v2/\*te-containers:DELETE",  "/traffic-engineering/api/topology/v2/\*te-containers:GET",  "/traffic-engineering/api/topology/v2/\*te-containers:PATCH",  "/traffic-engineering/api/topology/v2/\*te-containers:POST",  "/traffic-engineering/api/topology/v2/\*te-containers:PUT",  "/traffic-engineering/api/topology/v2/\*te-lsps:DELETE",  "/traffic-engineering/api/topology/v2/\*te-lsps:GET",  "/traffic-engineering/api/topology/v2/\*te-lsps:PATCH",  "/traffic-engineering/api/topology/v2/\*te-lsps:POST",  "/traffic-engineering/api/topology/v2/\*te-lsps:PUT",  "/traffic-engineering/api/topology/v2/\*vpn:DELETE",  "/traffic-engineering/api/topology/v2/\*vpn:GET",  "/traffic-engineering/api/topology/v2/\*vpn:POST",  "/traffic-engineering/api/topology/v2/\*vpn:PUT",  "/traffic-engineering/api/topology/v2:GET",  "/traffic-engineering/api/ui/config:GET",  "/traffic-engineering/api/ui/mapview/v1:GET",  "/traffic-engineering/api/ui/topology/v1:GET",  "/traffic-engineering/api/ui/topology/v1:POST",  "/ui/topology/v1/demands:GET",  "/ui/topology/v1/interfaces:GET",  "/ui/topology/v1/te-lsps:GET"  ]  },  {  "service": "ztpservice.v1.DeviceOnboardingEdgeService",  "methods": [  "Connect",  "NotifyZTPStatus"  ]  },  {  "service": "devicemanager.CommandService",  "methods": [  "RebootDevices"  ]  },  {  "service": "dpm.DpmService",  "methods": [  "BulkListImage",  "GetImage",  "ListImage",  "Subscribe",  "UpdateImage"  ]  }  ],  "unique\_identifier": "b1fcf6c7-cb82-4300-a53e-db2a42aec87c-1687517579",  "identity\_provider\_type": "Local"  },  "identity\_provider\_type": "Local"  } |

1. Use the id\_token in the REST header as ‘x-iam-token’ to access the API.

**Curl Examle:**

|  |
| --- |
| curl -X GET 'https://10.54.172.171/api/v2/config/device/' \  > -H 'content-type: application/json' \  > -H 'x-iam-token: eyJhbGci…' \  > --compressed \  > --insecure  [  "vmx106",  "ios-xr9",  "vmx104",  "vmx105",  "vmx103",  "vmx107",  "ios-xr8",  "vmx101",  "vmx102"  ] |

## Accessing API

The Paragon insight specific APIs are listed as part of every release. You can search for the APIs for your release at <https://www.juniper.net/documentation/product/us/en/paragon-automation/>.