



Restful API Guide

GAA10G10AC

06/2025

Release Version: 5.1.14

Copyright © 2025 Garland Technology, LLC. All rights reserved.

No part of this document may be reproduced in any form or by any means without prior written permission of Garland Technology, LLC.

The Garland Technology trademarks, service marks ("Marks") and other Garland Technology trademarks are the property of Garland Technology, LLC. PacketMAX Series products of marks are trademarks or registered trademarks of Garland Technology, LLC. You are not permitted to use these Marks without the prior written consent of Garland Technology.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Garland Technology and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute the warranty of any kind, express or implied.

Table of Contents

Restful API Basics	6
Login.....	6
Display Public Information	6
Display Key Press Timeout	6
Reset Key Press Timeout (60-3600 seconds)	6
Logout	6
System.....	7
Display System Information	7
Display General System Setting.....	7
Edit Chassis Name / Key Press Timeout	7
Display Authentication	7
Enable Primary TACACS.....	7
Disable Primary TACACS.....	7
Enable Secondary TACACS.....	8
Disable Secondary TACACS.....	8
Display Privilege	8
Display Group.....	8
Create Group.....	8
Modify Group.....	9
Delete Group.....	9
Display User.....	9
Create User	9
Change User Password.....	9
Delete User.....	10
Display Network Setting.....	10
Display Date & Time	10
Display Time Zone	10
Enable NTP No Authentication.....	10
Enable NTP Authentication	10
Disable NTP	10
Set Date/Time Manually	11

Set Network Setting	11
Display Syslog	11
Enable Syslog.....	11
Syslog Test	11
Disable Syslog.....	11
Display SNMP	11
Enable SNMPv2rw.....	12
Enable SNMPv3 MD5/DES	12
Enable SNMPv3 SHA/AES.....	12
SNMP Test	12
Disable SNMP	12
Reboot.....	12
Packet Broker	13
Display Packet Broker Configuration	13
Display Filter Template.....	13
Create Filter Template.....	13
Modify Filter Template.....	13
Delete Filter Template.....	14
Display Load Balance Group.....	14
Create Load Balance Group.....	14
Modify Load Balance Group.....	14
Delete Load Balance Group.....	14
Display Load Balance Policy	14
Modify Load Balance Policy	15
Display Config Map Counts	15
Create Config Map / Ingress Filter	15
Create Config Map / Ingress Filter / Egress Filter.....	16
Modify Config Map Priority.....	16
Enable/Disable Config Map.....	16
Delete Config Map	17
Clear Config Map Counters	17
Refresh Config Map Counters	17

Create I2GRE Encapsulate Tunnel – No VLAN	17
Delete I2GRE Encapsulate Tunnel – No VLAN	17
Create I2GRE Encapsulate Tunnel – VLAN	18
Delete I2GRE Encapsulate Tunnel – VLAN	18
Create I2GRE Decapsulate Tunnel – No VLAN	18
Delete I2GRE Decapsulate Tunnel – No VLAN	19
Create I2GRE Decapsulate Tunnel – VLAN	19
Delete I2GRE Decapsulate Tunnel – VLAN	19
Display Tunnel	19
Port Info	20
Display Port Configuration	20
Modify Port Configuration	20
Display Port Availability	20
Display Port Statistics	20
Clear Port Statistics	20

Restful API Basics

The restful API supports JSON over HTTPS. The restful API data types supported; GET, POST, and PUT.

This document contains restful API syntax examples that may be used as a reference. The specific restful API syntax may be captured using the GUI under the Developer Tools / Network / Headers and Preview.

Login

This example uses the default username and password, admin/gtadmin1. Upon successfully logging in the unit will return a Cookie / Session ID. The Session ID must be used as part of all additional commands sent to the unit.

POST <https://xxx.xxx.xxx.xxx/login>

Body Syntax Example

```
{"username":"admin","password":"gtadmin1","rememberme":true}
```

Example Session ID:

Set-Cookie: session=id=9iNS4gEJCOSQ9QcgVPd;path=/

Note: Once a session has been established it is controlled by the Key Press Timeout value, 60 to 3600 seconds. The Key Press Timeout value can be modified using the “Edit Chassis Name / Key Press Timeout” option in Section 2. System. The Key Press Timeout may be displayed and reset using the following options.

Display Public Information

GET <https://xxx.xxx.xxx.xxx/sysInfoPublic>

Display Key Press Timeout

GET <https://xxx.xxx.xxx.xxx/active>

Reset Key Press Timeout (60-3600 seconds)

PUT <https://xxx.xxx.xxx.xxx/miscCfg>

Body Syntax Example

```
{"chassisName":"System Test","keyPressTimeout":"3000"}
```

Logout

POST <https://xxx.xxx.xxx.xxx/logout>

System

Display System Information

GET <https://xxx.xxx.xxx.xxx/sysInfo>

Display General System Setting

GET <https://xxx.xxx.xxx.xxx/miscCfg>

Edit Chassis Name / Key Press Timeout

PUT <https://xxx.xxx.xxx.xxx/miscCfg>

Body Syntax Example

```
{"chassisName": "NewChassisName", "keyPressTimeout": "60"}
```

Note: Key press timeout range 60-3600 seconds.

Display Authentication

GET <https://xxx.xxx.xxx.xxx/authentication>

Enable Primary TACACS

PUT <https://xxx.xxx.xxx.xxx/authentication>

Body Syntax Example

```
{"local": true, "tacacs": {"on": true, "server": "192.168.1.166", "secret": "xyz", "timeout": "5"}, "tacacs2": {"on": false, "server": "10.10.10.200", "secret": "xyz", "timeout": "5"}}
```

Disable Primary TACACS

PUT <https://xxx.xxx.xxx.xxx/authentication>

Body Syntax Example

```
{"local": true, "tacacs": {"on": false, "server": "192.168.1.166", "secret": "xyz", "timeout": "5"}, "tacacs2": {"on": false, "server": "10.10.10.200", "secret": "xyz", "timeout": "5"}}
```

Enable Secondary TACACS

PUT <https://xxx.xxx.xxx.xxx/authentication>

Body Syntax Example

```
{
  "local": true,
  "tacacs": {
    "on": false,
    "server": "10.10.10.200",
    "secret": "xyz",
    "timeout": "5"
  },
  "tacacs2": {
    "on": true,
    "server": "192.168.1.173",
    "secret": "richardson1",
    "timeout": "5"
  }
}
```

Disable Secondary TACACS

PUT <https://xxx.xxx.xxx.xxx/authentication>

Body Syntax Example

```
{
  "local": true,
  "tacacs": {
    "on": false,
    "server": "10.10.10.200",
    "secret": "xyz",
    "timeout": "5"
  },
  "tacacs2": {
    "on": false,
    "server": "192.168.1.173",
    "secret": "richardson1",
    "timeout": "5"
  }
}
```

Display Privilege

GET <https://xxx.xxx.xxx.xxx/availablePrivileges>

Display Group

GET <https://xxx.xxx.xxx.xxx/groups>

Create Group

POST <https://xxx.xxx.xxx.xxx/groupAdd>

Body Syntax Example

```
{
  "name": "NewGroup",
  "privileges": [
    "AAA", "USR", "DTC", "DTV", "EXC", "IPC", "IPV", "LGC",
    "LGV", "MIS", "PBC", "PBV", "PTC", "PTV", "RBT", "TPC", "TPV", "UPG", "ADM"
  ]
}
```

Note: Privileges may be added or removed as desired.

AAA	authentication, authorization, account
ADM	user administrator
DTC	date, time, ntp configuration
DTV	date, time, ntp view
EXC	export/import
IPC	maintenance network ip configuration
IPV	maintenance network ip view
LGC	syslog, snmp configuration
LGV	syslog, snmp view

MIS	miscellaneous
PBC	packet broker configuration
PBV	packet broker view
PTC	port configuration
PTV	port view
RBT	chassis reboot
TPC	tap config
TPV	tap view
UPG	software field upgrade
USR	account configuration

Modify Group

POST <https://xxx.xxx.xxx.xxx/groupChange>

Body Syntax Example

```
{"name": "NewGroup", "privileges": ["MIS", "PBC", "PBV", "PTC"], "oldName": "NewGroup"}
```

Delete Group

POST <https://xxx.xxx.xxx.xxx/groupDelete>

Body Syntax Example

```
{"name": "NewGroup"}
```

Display User

GET <https://xxx.xxx.xxx.xxx/userAll>

Create User

POST <https://xxx.xxx.xxx.xxx/userAdd>

Body Syntax Example

```
{"username": "NewUser", "password": "NewUserPW", "group": "NewGroup"}
```

Change User Password

POST <https://xxx.xxx.xxx.xxx/userChange>

Body Syntax Example

```
{"username": "User123", "password": "NewPW", "group": "Group", "oldUsername": "User123"}
```

Delete User

POST <https://xxx.xxx.xxx.xxx/userDelete>

Body Syntax Example

```
{"username": "NewUser"}
```

Display Network Setting

GET <https://xxx.xxx.xxx.xxx/maintNetwork>

Display Date & Time

GET <https://xxx.xxx.xxx.xxx/dateTime>

Display Time Zone

GET <https://xxx.xxx.xxx.xxx/timezones>

Enable NTP No Authentication

PUT <https://xxx.xxx.xxx.xxx/dateTime>

Body Syntax Example

```
{"timeZone": "America/Chicago", "ntp": {"on": true, "ipAddress": "192.168.1.132", "authentication": "none", "authSym": {"keynum": "1", "keytype": "MD5", "key": "key"}}, "date": "5/30/2024", "ntpStatus": "off", "time": "10:51"}
```

Enable NTP Authentication

PUT <https://xxx.xxx.xxx.xxx/dateTime>

Body Syntax Example

```
{"timeZone": "America/Chicago", "ntp": {"on": true, "ipAddress": "192.168.1.132", "authentication": "symmetric", "authSym": {"keynum": "1", "keytype": "md5", "key": "%M5zLv16t~/CT\D<JI3`"}}, "date": "5/30/2024", "ntpStatus": "off", "time": "10:52"}
```

Disable NTP

PUT <https://xxx.xxx.xxx.xxx/dateTime>

Body Syntax Example

```
{"timeZone": "America/Chicago", "ntp": {"on": false, "usePool": false, "ipAddress": "xxx.xxx.xx.xxx"}, "date": "2/10/2021", "time": "15:33"}
```

Set Date/Time Manually

PUT <https://xxx.xxx.xxx.xxx/dateTime>

Body Syntax Example

```
{"timeZone": "America/Chicago", "ntp": {"on": false, "ipAddress": "127.0.0.1", "authentication": "none", "authSym": {"keynum": "1", "keytype": "MD5", "key": "key"}}, "date": "5/30/2024", "ntpStatus": "off", "time": "11:26"}
```

Set Network Setting

PUT <https://xxx.xxx.xxx.xxx/maintNetwork>

Body Syntax Example

```
{"ipv4Enable": true, "loadedSslCertUsed": false, "sslCertLoaded": false, "address": "192.168.1.28/24", "mask": "255.255.255.0", "gateway": "192.168.1.1", "ipv6Enable": false, "ipv6Address": "", "ipv6Gateway": "", "currentIpv4Address": "192.168.1.28", "currentIpv6Address": ""}
```

Display Syslog

GET <https://xxx.xxx.xxx.xxx/syslogCfg>

Enable Syslog

PUT <https://xxx.xxx.xxx.xxx/syslogCfg>

Body Syntax Example

```
{"on": true, "useUnitId": false, "unitId": "0", "ipAddress": "xxx.xxx.xxx.xxx", "protocol": "UDP", "port": "514"}
```

Syslog Test

POST <https://xxx.xxx.xxx.xxx/syslogTest>

Disable Syslog

PUT <https://xxx.xxx.xxx.xxx/syslogCfg>

Body Syntax Example

```
{"on": false, "useUnitId": false, "unitId": "0", "ipAddress": "xxx.xxx.xxx.xxx", "protocol": "UDP", "port": "514"}
```

Display SNMP

GET <https://xxx.xxx.xxx.xxx/snmpCfg>

Enable SNMPv2rw

PUT <https://xxx.xxx.xxx.xxx/snmpCfg>

Body Syntax Example

```
{"on":true,"accessPort":"161","trapPort":"162","trapIpAddress":"xxx.xxx.xxx.xxx","v3user":"user1234","v3authType":"MD5","v3authPass":"auth1234","v3privPass":"priv1234","v3privProt":"DES","communityPassword":"public","mode":"v2rw"}
```

Enable SNMPv3 MD5/DES

PUT <https://xxx.xxx.xxx.xxx/snmpCfg>

Body Syntax Example

```
{"on":true,"accessPort":"161","trapPort":"162","trapIpAddress":"xxx.xxx.xxx.xxx","v3user":"user1234","v3authType":"MD5","v3authPass":"auth1234","v3privPass":"priv1234","v3privProt":"DES","communityPassword":"public","mode":"v3"}
```

Enable SNMPv3 SHA/AES

PUT <https://xxx.xxx.xxx.xxx/snmpCfg>

Body Syntax Example

```
{"on":true,"accessPort":"161","trapPort":"162","trapIpAddress":"xxx.xxx.xxx.xxx","v3user":"user1234","v3authType":"SHA","v3authPass":"auth1234","v3privPass":"priv1234","v3privProt":"AES","communityPassword":"public","mode":"v3"}
```

SNMP Test

POST <https://xxx.xxx.xxx.xxx/snmpTest>

Disable SNMP

PUT <https://xxx.xxx.xxx.xxx/snmpCfg>

Body Syntax Example

```
{"on":false,"accessPort":"161","trapPort":"162","trapIpAddress":"xxx.xxx.xxx.xxx","v3user":"user","v3authType":"MD5","v3authPass":"auth1234","v3privPass":"prov1234","v3privProt":"DES","communityPassword":"gtpublic","mode":"v2rw"}
```

Reboot

POST <https://xxx.xxx.xxx.xxx/reboot>

Packet Broker

Display Packet Broker Configuration

GET <https://xxx.xxx.xxx.xxx/pbConfig>

Display Filter Template

GET <https://xxx.xxx.xxx.xxx/filtTemplates>

Create Filter Template

POST <https://xxx.xxx.xxx.xxx/filterTemplateAdd>

Body Syntax Example

```
{ "name": "Example", "description": "", "enabled": true, "filterType": "PASSBY", "srcMac": "", "srcMacMask": "ff:ff:ff:ff:ff:ff", "dstMac": "", "dstMacMask": "ff:ff:ff:ff:ff:ff", "etherType": "", "srcIp": "10.10.10.10", "srcIpMask": "255.255.255.255", "dstIp": "", "dstIpMask": "255.255.255.255", "srcIp6": "", "srcIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF", "dstIp6": "", "dstIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF", "innerVlanId": "", "outerVlanId": "", "dscp": "", "ipProtocol": "", "l4SrcStartPort": "", "l4SrcEndPort": "", "l4DstStartPort": "", "l4DstEndPort": "", "egressFilterPort": "" }
```

Note: The filter type may be PASSALL, PASSBY or DENYBY.

Modify Filter Template

POST <https://xxx.xxx.xxx.xxx/filterTemplateChange>

Body Syntax Example

```
{ "name": "Example", "description": "", "enabled": true, "filterType": "PASSBY", "dstMac": "", "dstMacMask": "ff:ff:ff:ff:ff:ff", "srcMac": "", "srcMacMask": "ff:ff:ff:ff:ff:ff", "etherType": "", "ipProtocol": "", "srcIp": "10.10.10.20", "srcIpMask": "255.255.255.255", "dstIp": "", "dstIpMask": "255.255.255.255", "l4SrcStartPort": "", "l4DstStartPort": "", "l4SrcEndPort": "", "l4DstEndPort": "", "outerVlanId": "", "innerVlanId": "", "dscp": "", "egressFilterPort": "", "srcIp6": "", "srcIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF", "dstIp6": "", "dstIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF", "oldName": "Example" }
```

Delete Filter Template

POST <https://xxx.xxx.xxx.xxx/filterTemplateRemove>

Body Syntax Example

```
{"name": "NewFilterTemp"}
```

Display Load Balance Group

GET <https://xxx.xxx.xxx.xxx/loadBalanceGroups>

Create Load Balance Group

POST <https://xxx.xxx.xxx.xxx/loadBalanceGroupAdd>

Body Syntax Example

```
{"name": "NewLBGroup", "description": "", "ports": ["17", "19", "23", "24"]}
```

Note: String all desired ports as shown in the example above.

Modify Load Balance Group

POST <https://xxx.xxx.xxx.xxx/loadBalanceGroupChange>

Body Syntax Example

```
{"name": "NewLBG", "description": "", "ports": ["21", "22", "23", "24", "26"],  
"oldName": "NewLBG"}
```

Delete Load Balance Group

POST <https://xxx.xxx.xxx.xxx/loadBalanceGroupRemove>

Body Syntax Example

```
{"name": "NewLBGroup"}
```

Display Load Balance Policy

GET <https://xxx.xxx.xxx.xxx/lbPolicy>

Modify Load Balance Policy

PUT <https://xxx.xxx.xxx.xxx/lbPolicy>

Body Syntax Example

```
{"ipv4Src": "YES", "ipv4Dst": "YES", "l4SrcPort": "NO", "l4DstPort": "NO", "macSrc": "NO", "macDst": "NO"}
```

Note: The Load Balance Policy allows L2 to be enabled or L3 and/or L4 to be enabled.

Display Config Map Counts

GET <https://xxx.xxx.xxx.xxx/configMapCounts>

Create Config Map / Ingress Filter

POST <https://xxx.xxx.xxx.xxx/configMapAdd>

Body Syntax Example

```
{"loadBalanceGroup": {}, "configMap": {"ingressPorts": ["03"], "egress": {"ports": ["04"], "trunkGroup": ""}, "name": "", "description": "", "enabled": true, "ingressFilters": [{"name": "Example", "description": "", "enabled": true, "filterType": "PASSBY", "srcMac": "", "srcMacMask": "ff:ff:ff:ff:ff:ff", "dstMac": "", "dstMacMask": "ff:ff:ff:ff:ff:ff", "etherType": "", "srcIp": "", "srcIpMask": "255.255.255.255", "dstIp": "10.10.10.10", "dstIpMask": "255.255.255.255", "srcIp6": "", "srcIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF", "dstIp6": "", "dstIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF", "innerVlanId": "", "outerVlanId": "", "dscp": "", "ipProtocol": "", "l4SrcStartPort": "", "l4SrcEndPort": "", "l4DstStartPort": "", "l4DstEndPort": "", "egressFilterPort": ""}]}, "egressFilters": {"04": []}}
```

Note: Multiple ingress ports, egress ports and ingress filters may be added.

Create Config Map / Ingress Filter / Egress Filter

POST <https://xxx.xxx.xxx.xxx/configMapAdd>

Body Syntax Example

```
{
  "loadBalanceGroup": {},
  "configMap": {
    "ingressPorts": ["03"],
    "egress": {
      "ports": ["04"],
      "trunkGroup": "",
      "name": "",
      "description": "",
      "enabled": true,
      "ingressFilters": [
        {
          "name": "Example",
          "description": "",
          "enabled": true,
          "filterType": "PASSBY",
          "srcMac": "",
          "srcMacMask": "ff:ff:ff:ff:ff:ff",
          "dstMac": "",
          "dstMacMask": "ff:ff:ff:ff:ff:ff",
          "etherType": "",
          "srcIp": "10.10.10.10",
          "srcIpMask": "255.255.255.255",
          "dstIp": "",
          "dstIpMask": "255.255.255.255",
          "srcIp6": "",
          "srcIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF",
          "dstIp6": "",
          "dstIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF",
          "innerVlanId": "",
          "outerVlanId": "",
          "dscp": "",
          "ipProtocol": "",
          "l4SrcStartPort": "",
          "l4SrcEndPort": "",
          "l4DstStartPort": "",
          "l4DstEndPort": "",
          "egressFilterPort": ""
        }
      ],
      "egressFilters": [
        {
          "name": "Example",
          "description": "",
          "enabled": true,
          "filterType": "PASSBY",
          "srcMac": "",
          "srcMacMask": "ff:ff:ff:ff:ff:ff",
          "dstMac": "",
          "dstMacMask": "ff:ff:ff:ff:ff:ff",
          "etherType": "",
          "srcIp": "",
          "srcIpMask": "255.255.255.255",
          "dstIp": "10.10.10.10",
          "dstIpMask": "255.255.255.255",
          "srcIp6": "",
          "srcIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF",
          "dstIp6": "",
          "dstIp6Mask": "FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF.FFFF",
          "innerVlanId": "",
          "outerVlanId": "",
          "dscp": "",
          "ipProtocol": "",
          "l4SrcStartPort": "",
          "l4SrcEndPort": "",
          "l4DstStartPort": "",
          "l4DstEndPort": "",
          "egressFilterPort": "04"
        }
      ]
    }
  }
}
```

Note: Multiple ingress ports, egress ports, ingress filters and egress filters may be added.

Modify Config Map Priority

POST <https://xxx.xxx.xxx.xxx/configMapPriorityEnableChange>

Body Syntax Example

```
[
  {
    "name": "CM1",
    "enabled": true
  },
  {
    "name": "CM2",
    "enabled": true
  },
  {
    "name": "CM3",
    "enabled": true
  }
]
```

Note: String the config maps as shown in the above example. The priority is established highest to lowest based on the order listed.

Enable/Disable Config Map

POST <https://xxx.xxx.xxx.xxx/configMapPriorityEnableChange>

Body Syntax Example

```
[
  {
    "name": "CM1",
    "enabled": true
  },
  {
    "name": "CM2",
    "enabled": false
  },
  {
    "name": "CM3",
    "enabled": true
  }
]
```

Note: String the config maps as shown in the above example. Config maps may be enabled or disabled by modifying the "enabled" option true/false, true = enabled, false = disabled.

Delete Config Map

POST <https://xxx.xxx.xxx.xxx/configMapRemove>

Body Syntax Example

```
["CM1", "CM2", "CM3"]
```

Note: Identify the config map(s) using the "name".

Clear Config Map Counters

POST <https://xxx.xxx.xxx.xxx/configMapCountsClear>

Refresh Config Map Counters

GET <https://xxx.xxx.xxx.xxx/configMapCounts>

Create I2GRE Encapsulate Tunnel – No VLAN

PUT <https://xxx.xxx.xxx.xxx/tunCreate>

Body Syntax Example

```
{"tunnel":{"access-port":5,"access-port-split":false,"access-split-part":"","tunnel-port":6,"tunnel-port-split":false,"tunnel-split-part":"","id":"","mac-address":"f2:93:c5:e5:30:13","next-hop-ip":"10.10.10.20","next-hop-mac":"f0:93:c5:a1:a1:a1","tunnel-ip":"10.10.10.10","tunnel-type":"gre-tx-only","udp-port":"","vnid":1234}}
```

Delete I2GRE Encapsulate Tunnel – No VLAN

PUT <https://xxx.xxx.xxx.xxx/tunDelete>

Body Syntax Example

```
{"tunnel":{"primaryVNID":1234,"secondaryVNIDs":[],"access-port":5,"access-port-split":false,"access-split-part":"","base-tunnel":true,"id":1,"mac-address":"f2:93:c5:e5:30:13","next-hop-ip":"10.10.10.20","next-hop-mac":"f0:93:c5:a1:a1:a1","tunnel-ip":"10.10.10.10","tunnel-port":6,"tunnel-port-split":false,"tunnel-split-part":"","tunnel-type":"gre-tx-only","udp-port":""}}
```

Create I2GRE Encapsulate Tunnel – VLAN

PUT <https://xxx.xxx.xxx.xxx/tunCreate>

Body Syntax Example

```
{"tunnel":{"access-port":5,"access-port-split":false,"access-split-part":"","tunnel-port":6,"tunnel-port-split":false,"tunnel-split-part":"","id":"","mac-address":"f2:93:c5:e5:30:13","next-hop-ip":"10.10.10.20","next-hop-mac":"f0:93:c5:a2:a2:a2","tunnel-ip":"10.10.10.10","tunnel-type":"gre-tx-only","tunnel-vlan-id":500,"udp-port":"","vnid":1234}}
```

Delete I2GRE Encapsulate Tunnel – VLAN

PUT <https://xxx.xxx.xxx.xxx/tunDelete>

Body Syntax Example

```
{"tunnel":{"primaryVNID":1234,"secondaryVNIDs":[],"access-port":5,"access-port-split":false,"access-split-part":"","base-tunnel":true,"id":1,"mac-address":"f2:93:c5:e5:30:13","next-hop-ip":"10.10.10.20","next-hop-mac":"f0:93:c5:a2:a2:a2","tunnel-ip":"10.10.10.10","tunnel-port":6,"tunnel-port-split":false,"tunnel-split-part":"","tunnel-type":"gre-tx-only","tunnel-vlan-id":500,"udp-port":""}}
```

Create I2GRE Decapsulate Tunnel – No VLAN

PUT <https://xxx.xxx.xxx.xxx/tunCreate>

Body Syntax Example

```
{"tunnel":{"access-port":6,"access-port-split":false,"access-split-part":"","tunnel-port":5,"tunnel-port-split":false,"tunnel-split-part":"","id":"","mac-address":"f0:93:c5:a1:a2:a3","next-hop-ip":"10.10.10.20","next-hop-mac":"","tunnel-ip":"10.10.10.10","tunnel-type":"gre-rx-only","udp-port":"","vnid":1234}}
```

Delete I2GRE Decapsulate Tunnel – No VLAN

PUT <https://xxx.xxx.xxx.xxx/tunDelete>

Body Syntax Example

```
{"tunnel":{"primaryVNID":1234,"secondaryVNIDs":[],"access-port":6,"access-port-split":false,"access-split-part":"","base-tunnel":true,"id":1,"mac-address":"f0:93:c5:a1:a2:a3","next-hop-ip":"10.10.10.20","next-hop-mac":"","tunnel-ip":"10.10.10.10","tunnel-port":5,"tunnel-port-split":false,"tunnel-split-part":"","tunnel-type":"gre-rx-only","udp-port":""}}
```

Create I2GRE Decapsulate Tunnel – VLAN

PUT <https://xxx.xxx.xxx.xxx/tunCreate>

Body Syntax Example

```
{"tunnel":{"access-port":6,"access-port-split":false,"access-split-part":"","tunnel-port":5,"tunnel-port-split":false,"tunnel-split-part":"","id":"","mac-address":"f2:93:c5:e5:30:13","next-hop-ip":"10.10.10.20","next-hop-mac":"","tunnel-ip":"10.10.10.10","tunnel-type":"gre-rx-only","tunnel-vlan-id":600,"udp-port":"","vnid":1234}}
```

Delete I2GRE Decapsulate Tunnel – VLAN

PUT <https://xxx.xxx.xxx.xxx/tunDelete>

Body Syntax Example

```
{"tunnel":{"primaryVNID":1234,"secondaryVNIDs":[],"access-port":6,"access-port-split":false,"access-split-part":"","base-tunnel":true,"id":1,"mac-address":"f2:93:c5:e5:30:13","next-hop-ip":"10.10.10.20","next-hop-mac":"","tunnel-ip":"10.10.10.10","tunnel-port":5,"tunnel-port-split":false,"tunnel-split-part":"","tunnel-type":"gre-rx-only","tunnel-vlan-id":600,"udp-port":""}}
```

Display Tunnel

GET <https://xxx.xxx.xxx.xxx/tunGet>

Port Info

Display Port Configuration

GET <https://xxx.xxx.xxx.xxx/portConfigGet>

Modify Port Configuration

PUT <https://xxx.xxx.xxx.xxx/portConfigPut>

Body Syntax Example

```
{
  "ports": [
    {
      "portNumber": "3",
      "speedSet": "1G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "4",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "5",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "6",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "7",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "8",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "9",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "10",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "11",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    },
    {
      "portNumber": "12",
      "speedSet": "10G",
      "mode": "normal",
      "description": "port description",
      "sfpVendorPn": "",
      "vlanIdTagAdd": 0,
      "vlanIdTagStrip": 0,
      "split": "NO"
    }
  ]
}
```

Note: Whenever modifying a port(s) all ports must be included in the body syntax.

Display Port Availability

GET <https://xxx.xxx.xxx.xxx/portAvailability>

Display Port Statistics

GET <https://xxx.xxx.xxx.xxx/portCounterGet>

Clear Port Statistics

PUT <https://xxx.xxx.xxx.xxx/portCounterClear>