

# SMxTAT2SA Series

## Unified API User Guide

33825 Rev. B

### Contents

|                                      |    |
|--------------------------------------|----|
| 1. Login .....                       | 3  |
| 2. Logout .....                      | 4  |
| 3. Reboot.....                       | 4  |
| 4. Get System Information .....      | 5  |
| 5. Set System Information.....       | 6  |
| 6. Get PoE Status.....               | 7  |
| 7. Get PoE Config.....               | 8  |
| 8. Set PoE Config .....              | 9  |
| 9. Get PoE Auto Power Reset .....    | 11 |
| 10. Set PoE Auto Power Reset.....    | 12 |
| 11. Get Port Statistics.....         | 14 |
| 12. Get Port Config.....             | 16 |
| 13. Set Port Config .....            | 17 |
| 14. Get SFP Port Detail .....        | 18 |
| 15. Firmware Upgrade .....           | 19 |
| 16. Get Firmware Upgrade Status..... | 20 |
| 17. Get Account Configuration.....   | 21 |
| 18. Set Account Configuration .....  | 21 |
| 19. Get MAC Table Information .....  | 22 |
| 20. Save Configuration .....         | 22 |
| 21. Get System Time .....            | 23 |
| 22. Set System Time .....            | 24 |
| 23. Get NTP Server.....              | 27 |
| 24. Set NTP Server.....              | 27 |
| 25. Get Syslog Server .....          | 28 |
| 26. Set Syslog Server .....          | 29 |
| 27. Get Syslog Log .....             | 30 |
| 28. Clear Syslog .....               | 30 |
| 29. Get Vlan Config .....            | 31 |
| 30. Set Vlan Config .....            | 32 |
| 31. Get Mac Based Vlan Config.....   | 34 |

|                                    |    |
|------------------------------------|----|
| 32. Get IP Address.....            | 35 |
| 33. Set IP Address.....            | 36 |
| 34. Get Mirror Config.....         | 37 |
| 35. Set Mirror Config.....         | 38 |
| 36. Cable Diagnostic.....          | 39 |
| 37. Get ACL Config .....           | 40 |
| 38. Set ACL Config .....           | 43 |
| 39. Create SSL Key.....            | 51 |
| 40. Get SSL Key Status.....        | 52 |
| 41. Device List Table.....         | 52 |
| 42. Get Config File List.....      | 54 |
| 43. Export Config.....             | 55 |
| 44. Import Config .....            | 56 |
| 45. Activate Config .....          | 57 |
| 46. Delete Config.....             | 58 |
| 47. Get Config Action Status ..... | 59 |
| 48. Ping.....                      | 60 |
| 49. Get Ping Status .....          | 61 |
| 50. Traceroute .....               | 62 |
| 51 Get Traceroute Status .....     | 63 |
| 52. Get SNMP Trap Config.....      | 64 |
| 53. Add SNMP Trap Config .....     | 65 |
| 54. Delete SNMP Trap Config.....   | 67 |
| cURL Commands v 1.7.....           | 68 |
| Record of Revisions .....          | 71 |

# 1. Login

**URL:** /api/login

**Method:** POST

**Request JSON:**

```
{
  "login": {
    "username" : "user123",
    "password" : "user123",
    "user_ip": "192.168.1.1",
    "sessid": "375118820"
  }
}
```

**Response JSON:**

```
{
  "response":{
    "status":"error",
    "message":"Wrong username or password!"
  }
}
```

**Section:**

| Name     | Data type | Allowed / Value   | Default Value |
|----------|-----------|-------------------|---------------|
| username | String    | 1-31 alphanumeric |               |
| password | String    | 0-31 alphanumeric |               |
| user_ip  | String    | <ip4 address>     |               |
| sessid   | String    | <cookie>          |               |

## 2. Logout

**URL:** /api/logout

**Method:** POST

**Request JSON:**

```
{
  "logout": {
    "sessid": "375118820"
  }
}
```

**Response JSON:**

```
{
  "response":{
    "status":"success"
  }
}
```

**Section:**

| Name   | Data type | Allowed / Value | Default Value |
|--------|-----------|-----------------|---------------|
| sessid | String    | <cookie>        |               |

## 3. Reboot

**URL:** /api/reboot

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "warm": "Yes"
  }
}
```

**Response JSON:** null

**Section:**

| Name | Data type | Allowed / Value | Default Value |
|------|-----------|-----------------|---------------|
| warm | String    | "Yes"           |               |

## 4. Get System Information

**URL:** /api/get\_sysinfo

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "information": {
      "model_name": "SM24TAT2SA",
      "description": "24-Port 1G Copper+2-Port Ethernet PoE Switch",
      "hardware_version": "v1.00",
      "mechanical_version": "v1.00",
      "firmware_version": "v1.02.1363",
      "mac_addr": "11-22-33-44-55-66",
      "serial_number": "152615261526",
      "system_name": "SM24TAT2SA",
      "location": "",
      "contact": "",
      "system_date": "2019-01-01 02:22:10 +0000",
      "uptime": "0 days, 2:22:25",
      "cpu_load": "18.7 %",
      "ram": {
        "total": "127520 KB",
        "free": "62996 KB"
      },
      "temperature_1": 50,
      "temperature_2": 64,
      "temperature_3": 43
    }
  }
}
```

## 5. Set System Information

**URL:** /api/set\_sysinfo

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "information": {
      "system_name": "SM24TAT2SA",
      "location": "Test Location",
      "contact": "Test Contact"
    }
  }
}
```

**Response JSON:**

```
{
  "system": {
    "information": {
      "system_name": "SM24TAT2SA",
      "location": "Test Location",
      "contact": "Test Contact"
    }
  }
}
```

**Section:**

| Name        | Data type | Allowed / Value    | Default Value |
|-------------|-----------|--------------------|---------------|
| system_name | String    | 0-128 alphanumeric |               |
| location    | String    | 0-128 alphanumeric |               |
| contact     | String    | 0-128 alphanumeric |               |

## 6. Get PoE Status

**URL:** /api/get\_poe\_status

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "poe": {
    "total_power_allocate": 308,
    "total_power_request": 308,
    "total_power_used": 92,
    "total_current_used": 178
  },
  "ports":[{
    "id": 1,
    "poe":{
      "pd_class": "0",
      "priority": "Low",
      "port_status": "PoE turned ON",
      "power_allocate": 154,
      "power_request": 154,
      "power_used": 44,
      "current_used": 86,
      "power_override": 0
    }
  },
  .....
]
```

**Section:**

| Name                 | Data type | Unit     |
|----------------------|-----------|----------|
| total_power_allocate | Integer   | 0.1 watt |
| total_power_request  | Integer   | 0.1 watt |
| total_power_used     | Integer   | 0.1 watt |
| total_current_used   | Integer   | mA       |
| power_allocate       | Integer   | 0.1 watt |
| power_request        | Integer   | 0.1 watt |
| power_used           | Integer   | 0.1 watt |

|                     |         |    |
|---------------------|---------|----|
| <b>current_used</b> | Integer | mA |
|---------------------|---------|----|

## 7. Get PoE Config

URL: /api/get\_poe\_config

Method: GET

Request JSON: null

Response JSON:

```
{
  "poe": {
    "poe_firmware_version": "208-188",
    "total_power_watts": 740,
    "power_determined_mode": "Class",
    "capacitor_detection": false,
    "profile_list": [
      {
        "id": 1,
        "name": "profile1"
      },
      .....
    ]
  },
  "ports": [
    {
      "id": 1,
      "poe": {
        "mode": "Enabled",
        "priority": "Low",
        "power_limit_user": 30,
        "schedule": "Disabled"
      }
    },
    .....
  ]
}
```

**Section:**

| Name                     | Data type | Unit | Note   |
|--------------------------|-----------|------|--|
| <b>total_power_watts</b> | Integer   | watt |  |
| <b>power_limit_user</b>  | Integer   | Watt |  |
| <b>Profile_list</b>      | Array     |      | <b>Ports[ ].poe.schedule</b> used to map PoE |



|  |  |  |                          |
|--|--|--|--------------------------|
|  |  |  | scheduling profile list. |
|--|--|--|--------------------------|

## 8. Set PoE Config

URL: /api/set\_poe\_config

Method: POST

Request JSON:

```
{
  "poe": {
    "power_determined_mode": "Class",
    "capacitor_detection": false
  },
  "ports": [
    {
      "id": 1,
      "poe": {
        "mode": "Enabled",
        "priority": "Low",
        "power_limit_user": 30,
        "schedule": "Disabled"
      }
    },
    .....
  ]
}
```

Response JSON:

```
{
  "poe": {
    "power_determined_mode": "Class",
    "capacitor_detection": false
  },
  "ports": [
    {
      "id": 1,
      "poe": {
        "mode": "Enabled",
        "priority": "Low",
        "power_limit_user": 30,
        "schedule": "Disabled"
      }
    }
  ]
}
```

```

    }
  },
  .....
]
}

```

**Section:**

| Name                         | Data type | Allowed / Value                     | Default Value |
|------------------------------|-----------|-------------------------------------|---------------|
| <b>power_determined_mode</b> | String    | "Class" 、 "Allocation" 、 "LLDP-Med" | Allocation    |
| <b>capacitor_detection</b>   | Boolean   |                                     | false         |
| <b>id</b>                    | Integer   | <Port number>                       |               |
| <b>mode</b>                  | String    | "Enabled" 、 "Disabled"              | Enabled       |
| <b>priority</b>              | String    | "Low" 、 "High" 、 "Critical"         | Low           |
| <b>power_limit_user</b>      | Integer   | 1-30 watt                           | 30            |
| <b>schedule</b>              | String    | "Disabled" 、 <Profile Name>         | Disabled      |

## 9. Get PoE Auto Power Reset

**URL:** /api/get\_poe\_auto\_reset

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "poe": {
    "auto_checking": false
  },
  "ports": [
    {
      "id": 1,
      "poe_auto_reset": {
        "ip": "0.0.0.0",
        "startup_time": 60,
        "interval_time": 30,
        "retry_time": 3,
        "error": 0,
        "total": 0,
        "failure_reboot": false,
        "reboot_time": 15,
        "max_reboot_times": 3
      }
    },
    ... ..
  ]
}
```

## 10. Set PoE Auto Power Reset

**URL:** /api/set\_poe\_auto\_reset

**Method:** POST

**Request JSON:**

```
{
  "poe": {
    "auto_checking": false
  },
  "ports": [
    {
      "id": 1,
      "poe_auto_reset": {
        "ip": "0.0.0.0",
        "startup_time": 60,
        "interval_time": 30,
        "retry_time": 3,
        "error": 0,
        "total": 0,
        "failure_reboot": false,
        "reboot_time": 15,
        "max_reboot_times": 3
      }
    },
    ... ..
  ]
}
```

**Response JSON:**

```
{
  "poe": {
    "auto_checking": false
  },
  "ports": [
    {
      "id": 1,
      "poe_auto_reset": {
        "ip": "0.0.0.0",
        "startup_time": 60,
        "interval_time": 30,
```

```

        "retry_time": 3,
        "error": 0,
        "total": 0,
        "failure_reboot": false,
        "reboot_time": 15,
        "max_reboot_times": 3
    }
},
... ..
]
}
    
```

**Section:**

| Name             | Data type | Allowed / Value | Default Value |
|------------------|-----------|-----------------|---------------|
| auto_checking    | Boolean   |                 | false         |
| id               | Integer   |                 | <Port number> |
| ip               | String    | <IPv4 Address>  | 0.0.0.0       |
| startup_time     | Integer   | 30-60 seconds   | 60            |
| interval_time    | Integer   | 10-120 seconds  | 30            |
| retry_time       | Integer   | 1-5 times       | 3             |
| failure_reboot   | Boolean   |                 | false         |
| reboot_time      |           | 3-120 seconds   | 15            |
| max_reboot_times |           | 0-10 times      | 0             |

## 11. Get Port Statistics

**URL:** /api/get\_port\_statistics

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "ports": [
    {
      "id": 1,
      "statistics": {
        "rx_packets": {
          "all": 93536,
          "octets": 11676072,
          "unicast": 44332,
          "multicast": 37536,
          "broadcast": 11672,
          "discard": 26816,
          "64 bytes": 55171,
          "65-127 bytes": 6235,
          "128-255 bytes": 5317,
          "256-511 bytes": 5841,
          "512-1023 bytes": 3493,
          "1024-1518 bytes": 1,
          "1519-max bytes": 0,
          "drop": 26816,
          "crc_alignment": 0,
          "oversize": 0,
          "undersize": 0,
          "fragments": 0,
          "jabber": 0
        },
        "tx_packets": {
          "all": 130311,
          "octets": 14036132,
          "unicast": 9516,
          "multicast": 1123,
          "broadcast": 119672,
          "discard": 0,

```

```
        "64 bytes": 77115,  
        "65-127 bytes": 9511,  
        "128-255 bytes": 336,  
        "256-511 bytes": 302,  
        "512-1023 bytes": 1251,  
        "1024-1518 bytes": 2668,  
        "1519-max bytes": 0,  
        "drop": 0,  
        "late_collision": 0,  
        "excessive_collision": 0  
    }  
},  
.....  
]  
}
```

## 12. Get Port Config

**URL:** /api/get\_port\_config

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "ports": [
    {
      "id": 1,
      "link": "1Gfdx",
      "media": "copper",
      "speed_mode": "Auto",
      "flow_control": false,
      "description": ""
    },
    ... ..
  ]
}
```



### 13. Set Port Config

**URL:** /api/set\_port\_config

**Method:** POST

**Request JSON:**

```
{
  "ports": [
    {
      "id": 1,
      "speed_mode": "Auto",
      "flow_control": false,
      "description": "test description"
    },
    ... ..
  ]
}
```

**Response JSON:**

```
{
  "ports": [
    {
      "id": 1,
      "link": "down",
      "media": "copper",
      "speed_mode": "Auto",
      "flow_control": false,
      "description": "test description"
    },
    ... ..
  ]
}
```

**Section:**

| Name       | Data type | Allowed / Value  | Default Value |
|------------|-----------|--|---------------|
| id         | Integer   | <Port number>  |               |
| speed_mode | String    | "Disabled"<br>"Auto"<br>"10Mbps HDX"<br>"10Mbps FDX"<br>"100Mbps HDX"<br>"100Mbps FDX" | Auto          |

|                     |         |                   |       |
|---------------------|---------|-------------------|-------|
|                     |         | "1Gbps FDX"       |       |
| <b>flow_control</b> | Boolean |                   | false |
| <b>description</b>  | String  | 0-63 alphanumeric |       |

## 14. Get SFP Port Detail

**URL:** /api/get\_sfp\_port\_detail

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "ports": [{
    "id": 1,
    "sfp":{
      "connector_type": "SFP or SFP Plus - LC",
      "fiber_type": "Multi-mode (MM)",
      "tx_central_wavelength": "850",
      "bit_rate": "1000 Mbps ",
      "vendor_oui": "00-17-2d",
      "vendor_name": "Axcen Photonics",
      "vendor_pn": "AXGD-5854-0511",
      "vendor_revision": "V2.0",
      "vendor_serial_number": "AX17200013613",
      "date_code": "170518",
      "temperature": "46.59 C",
      "vcc": "3.29 V",
      "mon1_bias": "5 mA",
      "mon2_tx_pwr": "-6.64 dBm",
      "mon3_rx_pwr": "none"
    }
  },
  ... ..
  ]
}
```

# 15. Firmware Upgrade

**URL:** /api/firmware\_upgrade

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "firmware": {
      "upgrade_url": "http://192.168.5.46/test.tar.gz"
    }
  }
}
```

---

```
{
  "system": {
    "firmware": {
      "upgrade_url": "tftp://192.168.5.46/test.tar.gz"
    }
  }
}
```

**Response JSON:** null (note: get action status used "[Get Firmware Upgrade Status](#)")

**Section:**

| Name        | Data type | Allowed / Value | Note                               |
|-------------|-----------|-----------------|------------------------------------|
| upgrade_url | String    | <URL>           | Support protocols :<br>tftp , http |

## 16. Get Firmware Upgrade Status

**URL:** /api/get\_firmware\_upgrade\_status

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "firmware": {
      "upgrade_status": "Error : The firmware is already update."
    }
  }
}
```

**Section:**

| Name           | Data type | Allowed / Value  |
|----------------|-----------|--|
| upgrade_status | String    | Never updated  |
|                |           | Downloading, please stand by...  |
|                |           | Processing, please stand by...   |
|                |           | PoE updating, please stand by...   |
|                |           | Erasing, please stand by...  |
|                |           | Flashing, please stand by...   |
|                |           | The device has been updated successfully.                                  |
|                |           | Error: Failed to downloaded the firmware.                                  |
|                |           | Error: The firmware is already update.                                     |
|                |           | Error: The firmware image is invalid. Please use a correct firmware image. |
|                |           | Error: Failed to upgraded the firmware.                                    |

## 17. Get Account Configuration

**URL:** /api/get\_account\_config

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "account": [{
    "username" : "admin",
    "privilege_level" : 15
  },
  ... ..
  ]
}
```

## 18. Set Account Configuration

**URL:** /api/set\_account\_config

**Method:** POST

**Request JSON:**

```
{
  "account": {
    "status" : "NEW",
    "username" : "superuser",
    "password" : "superuser",
    "privilege_level" : 15
  }
}
```

**Response JSON:**

```
{
  "account": [{
    "username" : "superuser",
    "privilege_level" : 15
  },
  ... ..
  ]
}
```

**Section:**

| Name            | Data type | Allowed / Value        | Default Value |
|-----------------|-----------|------------------------|---------------|
| status          | String    | "EDIT" 、 "NEW" 、 "DEL" |               |
| username        | String    | 1-31 alphanumeric      |               |
| password        | String    | 0-31 alphanumeric      |               |
| privilege_level | Integer   | 0-15                   | 0             |

**Note:** Only modify one at a time.

## 19. Get MAC Table Information

**URL:** /api/get\_dynamic\_mac\_table

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "mac_table": [{
    "type": "Dynamic",
    "mac": "11-22-33-44-55-66",
    "vid": 1,
    "port": 9
  },
  ... ..
  ]
}
```

## 20. Save Configuration

**URL:** /api/save\_configuration

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  " response ": {
    "status": "success",
    "message": "startup-config saved successfully."
  }
}
```

## 21. Get System Time

**URL:** /api/get\_system\_time

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "time": {
      "clock_source": "Local Setting",
      "system_date": "2017-01-01 01:01:30",
      "time_zone": "5400",
      "acronym": "",
      "daylight": {
        "mode": false,
        "offset": 60,
        "start_time": {
          "month": "Jan",
          "week": 1,
          "day": "Mon",
          "hour": 1
        },
        "end_time": {
          "month": "Jan",
          "week": 1,
          "day": "Mon",
          "hour": 1
        }
      }
    }
  }
}
```

## 22. Set System Time

**URL:** /api/set\_system\_time

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "time": {
      "clock_source": "Local Setting",
      "system_date": "2017-01-01 01:01:30",
      "time_zone": "5400",
      "acronym": "",
      "daylight": {
        "mode": false,
        "offset": 60,
        "start_time": {
          "month": "Jan",
          "week": 1,
          "day": "Mon",
          "hour": 1
        },
        "end_time": {
          "month": "Jan",
          "week": 1,
          "day": "Mon",
          "hour": 1
        }
      }
    }
  }
}
```

**Response JSON:**

```
{
  "system": {
    "time": {
      "clock_source": "Local Setting",
      "system_date": "2017-01-01 01:01:30",
      "time_zone": "5400",
      "acronym": "",

```



```

    "daylight":{
      "mode":false,
      "offset":60,
      "start_time": {
        "month": "Jan",
        "week": 1,
        "day": " Mon ",
        "hour": 1
      },
      "end_time": {
        "month": "Jan",
        "week": 1,
        "day": "Mon",
        "hour": 1
      }
    }
  }
}

```

**Section:**

| Name         | Data type | Allowed / Value  | Default Value |
|--------------|-----------|--|---------------|
| clock_source | String    | "Local Setting" 、 "NTP Server"   | Local Setting |
| system_date  | String    | "[Year]-[Month]-[Day] [Hour]:[Minute]:[Second]"  |               |
| time_zone    | String    | Reference "Time Zone Mapping Table"  |               |
| acronym      | String    | 0-16 alphanumeric  |               |
| mode         | Boolean   |  | False         |
| offset       | Integer   | 1-720 Min  | 60            |
| month        | String    | "Jan" 、 "Feb" 、 "Mar"<br>"Apr" 、 "May" 、 "Jun"<br>"Jul" 、 "Aug" 、 "Sep"<br>"Oct" 、 "Nov" 、 "Dec" | Jan           |
| week         | Integer   | 1-5  | 1             |
| day          | String    | "Mon" 、 "Tue" 、 "Wed"<br>"Thu" 、 "Fri" 、 "Sat" 、 "Sun"   | Mon           |
| hour         | Integer   | 0-23   | 0             |

**Time Zone Mapping Table:**

| <b>Value</b> | <b>Note</b> |
|--------------|-------------|
| -7200        | (GMT-12:00) |
| -6600        | (GMT-11:00) |
| -6000        | (GMT-10:00) |
| -5400        | (GMT-09:00) |
| -4800        | (GMT-08:00) |
| -4200        | (GMT-07:00) |
| -3600        | (GMT-06:00) |
| -3000        | (GMT-05:00) |
| -2700        | (GMT-04:30) |
| -2400        | (GMT-04:00) |
| -2100        | (GMT-03:30) |
| -1800        | (GMT-03:00) |
| -1200        | (GMT-02:00) |
| -600         | (GMT-01:00) |
| 0            | (GMT+00:00) |
| 600          | (GMT+01:00) |
| 1200         | (GMT+02:00) |
| 1800         | (GMT+03:00) |
| 2100         | (GMT+03:30) |
| 2400         | (GMT+04:00) |
| 2700         | (GMT+04:30) |
| 3000         | (GMT+05:00) |
| 3300         | (GMT+05:30) |
| 3450         | (GMT+05:45) |
| 3600         | (GMT+06:00) |
| 3900         | (GMT+06:30) |
| 4200         | (GMT+07:00) |
| 4800         | (GMT+08:00) |
| 5400         | (GMT+09:00) |
| 5700         | (GMT+09:30) |
| 6000         | (GMT+10:00) |
| 6600         | (GMT+11:00) |
| 7200         | (GMT+12:00) |

## 23. Get NTP Server

**URL:** /api/get\_ntp\_server

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "ntp": {
      "server": [
        {"id": 1, "host": "192.168.1.1"},
        ... ..
      ],
      "interval": 60
    }
  }
}
```

## 24. Set NTP Server

**URL:** /api/set\_ntp\_server

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "ntp": {
      "server": [
        {"id": 1, "host": "192.168.1.1"},
        ... ..
      ],
      "interval": 60
    }
  }
}
```

**Response JSON:**

```
{
  "system": {
    "ntp": {
      "server": [
        {"id": 1, "host": "192.168.1.1"},
```

```

        ... ..
    ],
    "interval":60
  }
}

```

**Section:**

| Name         | Data type | Allowed / Value                 | Default Value |
|--------------|-----------|---------------------------------|---------------|
| id           | Integer   | 1-6                             |               |
| host         | String    | <IPv4 address>                  |               |
| ntp_interval | Integer   | 5 、 10 、 15 、 30 、 60 、 120 min | 60            |

## 25. Get Syslog Server

**URL:** /api/get\_syslog\_server

**Method:** GET

**Request JSON:** null

**Response JSON:**

```

{
  "event_notification":{
    "syslog":{
      "mode": false,
      "server":[
        {"id": 1,"address":"192.168.1.1"},
        ... ..
      ]
    }
  }
}

```

## 26. Set Syslog Server

**URL:** /api/set\_syslog\_server

**Method:** POST

**Request JSON:**

```
{
  "event_notification":{
    "syslog":{
      "mode":false,
      "server":[
        {"id":1,"address":"192.168.1.1"},
        ....
      ]
    }
  }
}
```

**Response JSON:**

```
{
  "event_notification":{
    "syslog":{
      "mode":false,
      "server":[
        {"id":1,"address":"192.168.1.1"},
        ....
      ]
    }
  }
}
```

**Section:**

| Name    | Data type | Allowed / Value | Default Value |
|---------|-----------|-----------------|---------------|
| mode    | Boolean   |                 | false         |
| id      | Integer   | 1-6             |               |
| address | String    | <IPv4 address>  |               |

## 27. Get Syslog Log

**URL:** /api/get\_syslog

**Method:** GET

**Request JSON:** null

**Response JSON:**

(Note: Only get the latest 100 entries.)

```
{
  "system": {
    "syslog": {
      "log": [{
        "id": 1,
        "level": "Info",
        "time": "2017-01-01T17:14:12+0000",
        "message": " Login passed for user 'admin' through HTTP from 192.168.0.46 and
authenticated by local method"
      },
      ... ..
    ]
  }
}
```

## 28. Clear Syslog

**URL:** /api/clear\_syslog

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "syslog": {
      "log": []
    }
  }
}
```

## 29. Get Vlan Config

**URL:** /api/get\_vlan\_config

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "vlan": {
    "allowed_access_vlans": "1",
    "ethertype_custom_s_ports": "88a8"
  },
  "ports": [{
    "id": 1,
    "vlan": {
      "mode": "Access",
      "access": {
        "pvid": 1
      },
      "trunk": {
        "pvid": 1,
        "egress_tagging": "Untag Port VLAN",
        "allowed_vlan": "1"
      },
      "hybrid": {
        "pvid": 1,
        "port_type": "C-Port",
        "ingress_filter": false,
        "ingress_accept": "Tagged and Untagged",
        "egress_tagging": "Untag Port VLAN",
        "allowed_vlan": "1"
      }
    }
  }
],
  ...
}
```

## 30. Set Vlan Config

**URL:** /api/set\_vlan\_config

**Method:** POST

**Request JSON:**

```
{
  "vlan": {
    "allowed_access_vlans": "1",
    "ethertype_custom_s_ports": "88a8"
  },
  "ports": [{
    "id": 2,
    "vlan": {
      "mode": "Access",
      "access": {
        "pvid": 1
      }
    }
  }, {
    "id": 3,
    "vlan": {
      "mode": "Trunk",
      "trunk": {
        "pvid": 1,
        "egress_tagging": "Untag Port VLAN",
        "allowed_vlan": "1"
      }
    }
  }, {
    "id": 4,
    "vlan": {
      "mode": "Hybrid",
      "hybrid": {
        "pvid": 1,
        "port_type": "C-Port",
        "ingress_filter": false,
        "ingress_accept": "Tagged and Untagged",
        "egress_tagging": "Untag Port VLAN",
        "allowed_vlan": "1"
      }
    }
  }
}
```



```

        }
    }
},
.....
]
}

```

**Response JSON:**

```

{
  "vlan": {
    "allowed_access_vlans": "1",
    "ethertype_custom_s_ports": "88a8"
  },
  "ports": [{
    "id": 1,
    "vlan": {
      "mode": "Access",
      "access": {
        "pvid": 1
      },
      "trunk": {
        "pvid": 1,
        "egress_tagging": "Untag Port VLAN",
        "allowed_vlan": "1"
      },
      "hybrid": {
        "pvid": 1,
        "port_type": "C-Port",
        "ingress_filter": false,
        "ingress_accept": "Tagged and Untagged",
        "egress_tagging": "Untag Port VLAN",
        "allowed_vlan": "1"
      }
    }
  ]
},
.....
]
}

```

**Section:**

| Name                          | Data type | Allowed / Value   | Default Value          |
|-------------------------------|-----------|---|------------------------|
| allowed_access_vlans          | String    | <vlan-list>   | 1                      |
| ethertype_custom_s_ports      | String    | <Ethertype>   | 88a8                   |
| id                            | Integer   | <Port number>   |                        |
| mode                          | String    | "Access" 、 "Trunk" 、 "Hybrid"                             | Access                 |
| pvid                          | Integer   | 1-4095  | 1                      |
| port_type                     | String    | "UNAWARE"<br>"C-Port"<br>"S-Port"<br>"S-Custom-Port"      | C-Port                 |
| ingress_filter                | Boolean   |   | false                  |
| ingress_accept                | String    | "Tagged and Untagged"<br>"Tagged only"<br>"Untagged only" | Tagged and<br>Untagged |
| egress_tagging<br>(in trunk)  | String    | "Untag Port VLAN"<br>"Tag All"                            | Untag Port VLAN        |
| egress_tagging<br>(in hybrid) | String    | "Untag Port VLAN"<br>"Tag All"<br>"Untag All"             | Untag Port VLAN        |
| allowed_vlan                  | String    | <vlan-list>   | 1                      |

## 31. Get Mac Based Vlan Config

**URL:** /api/get\_mac\_based\_vlan

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "vlan":{
    "mac_based_vlan": [{
      "mac": "00-11-22-33-44-55",
      "vid": 15
    }
    ... ..
  ]
}
```

## 32. Get IP Address

**URL:** /api/get\_ip\_address

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "ip": {
      "interfaces": [{
        "vid": 1,
        "ipv4": {
          "dhcp": false,
          "fallback": 0,
          "dhcp_addr": "",
          "static_addr": "192.168.111.126",
          "static_mask": 24
        },
        "ipv6": {
          "static_addr": "",
          "static_mask": 0
        }
      }
      ... ..
    ]
  }
}
```

## 33. Set IP Address

**URL:** /api/set\_ip\_address

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "ip": {
      "interfaces": [{
        "vid": 1,
        "ipv4": {
          "dhcp": false,
          "fallback": 0,
          "static_addr": "192.168.111.126",
          "static_mask": 24
        },
        "ipv6": {
          "static_addr": "",
          "static_mask": 0
        }
      }
      ... ..
    ]
  }
}
```

**Response JSON:**

```
{
  "system": {
    "ip": {
      "interfaces": [{
        "vid": 1,
        "ipv4": {
          "dhcp": false,
          "fallback": 0,
          "dhcp_addr": "",
          "static_addr": "192.168.111.126",
          "static_mask": 24
        },
      }
    ]
  }
}
```

```

    "ipv6": {
        "static_addr": "",
        "static_mask": 0
    }
}
...
]
}
}
}

```

**Section:**

| Name              | Data type | Allowed / Value | Default Value |
|-------------------|-----------|-----------------|---------------|
| dhcp              | Boolean   |                 |               |
| fallback          | Integer   | 1-4294967295    |               |
| ipv4: static_addr | String    | <ipv4 address>  |               |
| ipv4: static_mask | Integer   | 1-30            |               |
| ipv6: static_addr | String    | <ipv6 address>  |               |
| ipv6: static_mask | Integer   | 1-128           |               |

### 34. Get Mirror Config

URL: /api/get\_mirror\_config

Method: GET

Request JSON: null

Response JSON:

```

{
  "system": {
    "mirror": [{
      "session": 1,
      "enable": false,
      "destination_port": 2,
      "source_tx": "4,6-8",
      "source_rx": "3,5,7-8"
    }]
  }
}

```

**Note:** This configuration is based on the SMxTAT2SA chip function.

## 35. Set Mirror Config

**URL:** /api/set\_mirror\_config

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "mirror": [{
      "session": 1,
      "enable": false,
      "destination_port": 2,
      "source_tx": "4,6-8",
      "source_rx": "3,5,7-8"
    }]
  }
}
```

**Response JSON:**

```
{
  "system": {
    "mirror": [{
      "session": 1,
      "enable": false,
      "destination_port": 2,
      "source_tx": "4,6-8",
      "source_rx": "3,5,7-8"
    }]
  }
}
```

**Section:**

| Name             | Data type | Allowed / Value | Default Value |
|------------------|-----------|-----------------|---------------|
| session          | Integer   | 1               | 1             |
| enable           | Boolean   |                 | false         |
| destination_port | Integer   | <port number>   | 1             |
| source_tx        | String    | <port list>     |               |
| source_rx        | String    | <port list>     |               |

**Note:** This configuration is based on the SMxTAT2SA chip function.

## 36. Cable Diagnostic

**URL:** /api/cable\_diagnostics

**Method:** POST

**Request JSON:**

```
{
  "cable": {
    "port": 5
  }
}
```

**Response JSON:**

```
{
  "ports": [
    "id": 7,
    "cable_diagnostic": {
      "link": "1G",
      "result": "OK",
      "length": "6.00 (m)"
    }
  ]
}
```

**Section:**

| Name | Data type | Allowed / Value | Default Value |
|------|-----------|-----------------|---------------|
| port | Integer   | <port number>   |               |

**Note:** This configuration is based on the SMxTAT2SA chip function.

## 37. Get ACL Config

**URL:** /api/get\_acl\_config

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "acl": [
    {
      "ace_id": 1,
      "ingress_port_list": "1",
      "frame_type": "Any",
      "action": "Deny",
      "mirror": true,
      "counter": true,
      "metering": {
        "enable": true,
        "bandwidth": 900000
      }
    },
    {
      "ace_id": 2,
      "ingress_port_list": "1",
      "frame_type": "Ethernet Type",
      "action": "Deny",
      "mirror": true,
      "counter": true,
      "metering": {
        "enable": true,
        "bandwidth": 900000
      },
      "ethernet_type": {
        "mac": {
          "smac_filter": "Specific",
          "smac_value": "00-00-00-00-00-01",
          "dmac_filter": "Specific",
          "dmac_value": "00-00-00-00-00-02"
        },
        "ether_type": {
```



```
    "filter": "Specific",
    "value": "aaaa"
  },
  "vlan": {
    "ctag": {
      "tagged": "Enabled",
      "vid_filter": "Specific",
      "vid": 101,
      "priority": "4-7"
    },
    "stag": {
      "tagged": "Enabled",
      "vid_filter": "Specific",
      "vid": 102,
      "priority": "4-7"
    }
  }
}
},
{
  "ace_id": 3,
  "ingress_port_list": "1",
  "frame_type": "IPv4",
  "action": "Deny",
  "mirror": true,
  "counter": true,
  "metering": {
    "enable": true,
    "bandwidth": 900000
  },
  "ipv4": {
    "protocol_filter": "TCP",
    "protocol_value": 6,
    "fragment": "No",
    "tos_filter": "Any",
    "tos_dhcp_value": "",
    "tos_ip_value": "",
    "sip_filter": "Any",
    "sip_value": "0.0.0.0",
```

```
"sip_mask": 0,
"dip_filter": "Any",
"dip_value": "0.0.0.0",
"dip_mask": 0,
"icmp": {
  "type_filter": "Any",
  "type_value": 255,
  "code_filter": "Any",
  "code_value": 255
},
"udp": {
  "sport_filter": "Any",
  "sport_low": 0,
  "sport_high": 65535,
  "dport_filter": "Any",
  "dport_low": 0,
  "dport_high": 65535
},
"tcp": {
  "sport_filter": "Any",
  "sport_low": 0,
  "sport_high": 65535,
  "dport_filter": "Any",
  "dport_low": 0,
  "dport_high": 65535,
  "tcp_fin": "Any",
  "tcp_syn": "Any",
  "tcp_rst": "Any",
  "tcp_psh": "Any",
  "tcp_ack": "Any",
  "tcp_urg": "Any"
}
}
}
... ..
]
```

**Note:** This configuration is based on the SMxTAT2SA chip function.

## 38. Set ACL Config

**URL:** /api/set\_acl\_config

**Method:** POST

**Request JSON:**

```
{
  "acl": {
    "ace_id": 1,
    "next_ace_id": 0,
    "ingress_port_list": "1",
    "frame_type": "Any",
    "action": "Deny",
    "mirror": true,
    "counter": true,
    "metering": {
      "enable": true,
      "bandwidth": 900000
    }
  }
}
```

```
{
  "acl": {
    "ace_id": 2,
    "next_ace_id": 0,
    "ingress_port_list": "1",
    "frame_type": "Ethernet Type",
    "action": "Deny",
    "mirror": true,
    "counter": true,
    "metering": {
      "enable": true,
      "bandwidth": 900000
    },
    "ethernet_type": {
      "mac": {
        "smac_filter": "Specific",
        "smac_value": "00-00-00-00-00-01",
        "dmac_filter": "Specific",
        "dmac_value": "00-00-00-00-00-02"
      }
    }
  }
}
```

```

    },
    "ether_type": {
        "filter": "Specific",
        "value": "aaaa"
    },
    "vlan": {
        "ctag": {
            "tagged": "Enabled",
            "vid_filter": "Specific",
            "vid": 101,
            "priority": "4-7"
        },
        "stag": {
            "tagged": "Enabled",
            "vid_filter": "Specific",
            "vid": 102,
            "priority": "4-7"
        }
    }
}

```

```

{
  "acl": {
    "ace_id": 3,
    "next_ace_id": 0,
    "ingress_port_list": "1",
    "frame_type": "IPv4",
    "action": "Deny",
    "mirror": true,
    "counter": true,
    "metering": {
      "enable": true,
      "bandwidth": 900000
    },
    "ipv4": {
      "protocol_filter": "TCP",
      "protocol_value": 6,
      "fragment": "No",

```

```
"tos_filter": "Any",
"tos_dscp_value": "",
"tos_ip_value": "",
"sip_filter": "Any",
"sip_value": "0.0.0.0",
"sip_mask": 0,
"dip_filter": "Any",
"dip_value": "0.0.0.0",
"dip_mask": 0,
"icmp": {
  "type_filter": "Any",
  "type_value": 255,
  "code_filter": "Any",
  "code_value": 255
},
"udp": {
  "sport_filter": "Any",
  "sport_low": 0,
  "sport_high": 65535,
  "dport_filter": "Any",
  "dport_low": 0,
  "dport_high": 65535
},
"tcp": {
  "sport_filter": "Any",
  "sport_low": 0,
  "sport_high": 65535,
  "dport_filter": "Any",
  "dport_low": 0,
  "dport_high": 65535,
  "tcp_fin": "Any",
  "tcp_syn": "Any",
  "tcp_rst": "Any",
  "tcp_psh": "Any",
  "tcp_ack": "Any",
  "tcp_urg": "Any"
}
}
```

```
}
```

**Response JSON:**

```
{
  "acl": [
    {
      "ace_id": 1,
      "ingress_port_list": "1",
      "frame_type": "Any",
      "action": "Deny",
      "mirror": true,
      "counter": true,
      "metering": {
        "enable": true,
        "bandwidth": 900000
      }
    },
    {
      "ace_id": 2,
      "ingress_port_list": "1",
      "frame_type": "Ethernet Type",
      "action": "Deny",
      "mirror": true,
      "counter": true,
      "metering": {
        "enable": true,
        "bandwidth": 900000
      },
      "ethernet_type": {
        "mac": {
          "smac_filter": "Specific",
          "smac_value": "00-00-00-00-00-01",
          "dmac_filter": "Specific",
          "dmac_value": "00-00-00-00-00-02"
        },
        "ether_type": {
          "filter": "Specific",
          "value": "aaaa"
        }
      },
      "vlan": {
```

```
    "ctag": {
      "tagged": "Enabled",
      "vid_filter": "Specific",
      "vid": 101,
      "priority": "4-7"
    },
    "stag": {
      "tagged": "Enabled",
      "vid_filter": "Specific",
      "vid": 102,
      "priority": "4-7"
    }
  }
},
{
  "ace_id": 3,
  "ingress_port_list": "1",
  "frame_type": "IPv4",
  "action": "Deny",
  "mirror": true,
  "counter": true,
  "metering": {
    "enable": true,
    "bandwidth": 900000
  },
  "ipv4": {
    "protocol_filter": "TCP",
    "protocol_value": 6,
    "fragment": "No",
    "tos_filter": "Any",
    "tos_dhcp_value": "",
    "tos_ip_value": "",
    "sip_filter": "Any",
    "sip_value": "0.0.0.0",
    "sip_mask": 0,
    "dip_filter": "Any",
    "dip_value": "0.0.0.0",
    "dip_mask": 0,
```

```

    "icmp": {
      "type_filter": "Any",
      "type_value": 255,
      "code_filter": "Any",
      "code_value": 255
    },
    "udp": {
      "sport_filter": "Any",
      "sport_low": 0,
      "sport_high": 65535,
      "dport_filter": "Any",
      "dport_low": 0,
      "dport_high": 65535
    },
    "tcp": {
      "sport_filter": "Any",
      "sport_low": 0,
      "sport_high": 65535,
      "dport_filter": "Any",
      "dport_low": 0,
      "dport_high": 65535,
      "tcp_fin": "Any",
      "tcp_syn": "Any",
      "tcp_rst": "Any",
      "tcp_psh": "Any",
      "tcp_ack": "Any",
      "tcp_urg": "Any"
    }
  }
}
... ..
]
}

```

**Section:**

| Name              | Data type | Allowed / Value                  | Default Value |
|-------------------|-----------|----------------------------------|---------------|
| ace_id            | Integer   | 1-256                            |               |
| next_ace_id       | Integer   | 0-256                            | 0             |
| ingress_port_list | String    | "Any" 、 <port number>            | Any           |
| frame_type        | String    | "Any" 、 "Ethernet Type" 、 "IPv4" | Any           |



|                      |         |  |         |
|----------------------|---------|--|---------|
| action               | String  | "Deny" 、 "Permit" 、 "Shutdown"   | Permit  |
| mirror               | Boolean |  | false   |
| counter              | Boolean |  | false   |
| metering : enable    | Boolean |  | false   |
| metering : bandwidth | Integer | 16-1000000   | 1000000 |
| mac : smac_filter    | String  | "Any" 、 "Specific"   | Any     |
| mac : smac_value     | String  | <mac address>  |         |
| mac : smac_filter    | String  | "Any" 、 "Specific" 、<br>"MC" 、 "BC" 、 "UC"   | Any     |
| mac : smac_value     | String  | <mac address>  |         |
| ether_type : filter  | String  | "Any" 、 "Specific"   | Any     |
| ether_type : value   | String  | 0000-FFFF  | FFFF    |
| tagged               | String  | "Any" 、 "Disabled" 、 "Enabled"   | Any     |
| vid_filter           | String  | "Any" 、 "Specific"   | Any     |
| vid                  | Integer | 1-4095   |         |
| priority             | String  | "0" 、 "1" 、 "2" 、 "3" 、<br>"4" 、 "5" 、 "6" 、 "7" 、 "0-1" 、 "2-3" 、<br>"4-5" 、 "6-7" 、 "0-3" 、 "4-7" 、 "Any"  | Any     |
| protocol_filter      | String  | "Any" 、 "ICMP" 、 "UDP" 、<br>"TCP" 、 "Other"  | Any     |
| protocol_value       | Integer | 0-255  |         |
| fragment             | String  | "Any" 、 "Yes" 、 "No"   | Any     |
| tos_filter           | String  | "Any" 、 "DSCP" 、 "IP Precedence"   | Any     |
| tos_dscp_value       | String  | "0(BE)" 、 "1" 、 "2" 、 "3" 、 "4" 、 "5" 、<br>"6" 、 "7" 、 "8(CS1)" 、 "9" 、<br>"10(AF11)" 、 "11" 、 "12(AF12)" 、<br>"13" 、 "14(AF13)" 、 "15" 、<br>"16(CS2)" 、 "17" 、 "18(AF21)" 、<br>"19" 、 "20(AF22)" 、 "21" 、<br>"22(AF23)" 、 "23" 、 "24(CS3)" 、<br>"25" 、 "26(AF31)" 、 "27" 、<br>"28(AF32)" 、 "29" 、 "30(AF33)" 、<br>"31" 、 "32(CS4)" 、 "33" 、<br>"34(AF41)" 、 "35" 、 "36(AF42)" 、<br>"37" 、 "38(AF43)" 、 "39" 、<br>"40(CS5)" 、 "41" 、 "42" 、 "43" 、<br>"44" 、 "45" 、 "46(EF)" 、 "47" 、<br>"48(CS6)" 、 "49" 、 "50" 、 "51" 、<br>"52" 、 "53" 、 "54" 、 "55" 、 |         |

|                    |         |   |     |
|--------------------|---------|---|-----|
|                    |         | "56(CS7)"、"57"、"58"、"59"、<br>"60"、"61"、"62"、"63"                            |     |
| tos_ip_value       | String  | "0"、"1"、"2"、"3"、"4"、"5"、<br>"6"、"7"、"0-1"、"2-3"、"4-5"、<br>"6-7"、"0-3"、"4-7" |     |
| sip_filter         | String  | "Any"、"Host"、"Network"  | Any |
| sip_value          | String  | <ip address>  |     |
| sip_mask           | Integer | 1-32  |     |
| dip_filter         | String  | "Any"、"Host"、"Network"  | Any |
| dip_value          | String  | <ip address>  |     |
| dip_mask           | Integer | 1-32  |     |
| icmp : type_filter | String  | "Any"、"Specific"  | Any |
| icmp : type_value  | Integer | 0-255   |     |
| icmp : code_filter | String  | "Any"、"Specific"  | Any |
| icmp : code_value  | Integer | 0-255   |     |
| sport_filter       | String  | "Any"、"Specific"、"Range"  | Any |
| sport_low          | Integer | 0-65535   |     |
| sport_high         | Integer | 0-65535   |     |
| dport_filter       | String  | "Any"、"Specific"、"Range"  | Any |
| dport_low          | Integer | 0-65535   |     |
| dport_high         | Integer | 0-65535   |     |
| tcp_fin            | String  | "0"、"1"、"Any"   | Any |
| tcp_syn            | String  | "0"、"1"、"Any"   | Any |
| tcp_rst            | String  | "0"、"1"、"Any"   | Any |
| tcp_psh            | String  | "0"、"1"、"Any"   | Any |
| tcp_ack            | String  | "0"、"1"、"Any"   | Any |
| tcp_urg            | String  | "0"、"1"、"Any"   | Any |

**Note 1:** This configuration is based on the SMxTAT2SA chip function.

**Note 2:** Only modify one ACL entry at a time.

**Note 3:** The "frame\_type" determines the set JOSN format in ACL entry.

**Note 4:** The ACL entry set at the end of the table when "next\_ace\_id" value is 0.

## 39. Create SSL Key

**URL:** /api/create\_ssl\_key

**Method:** POST

**Request JSON:**

```
{
  "ssl_key": {
    "create": {
      "country": "GB",
      "state": "Test State",
      "location": "Test Locality",
      "organization": "Organization Name",
      "organizational_unit": "Organizational Unit Name",
      "common_name": "Common Name",
      "email": "test@example.com",
      "days": 3650,
      "key_length": 2048
    }
  }
}
```

**Response JSON:**

```
{
  "response": {
    "status": "success"
  }
}
```

**Section:**

| Name                | Data type | Allowed / Value    | Default Value |
|---------------------|-----------|--------------------|---------------|
| country             | String    | 0-2 alphanumeric   |               |
| state               | String    | 0-128 alphanumeric |               |
| location            | String    | 0-128 alphanumeric |               |
| organization        | String    | 0-64 alphanumeric  |               |
| organizational_unit | String    | 0-64 alphanumeric  |               |
| common_name         | String    | 0-64 alphanumeric  |               |
| email               | String    | 0-64 alphanumeric  |               |
| days                | Integer   | 1-3650 days        | 3650          |
| key_length          | Integer   | 2048 bits          | 2048          |

**Note:** When creating an SSL Key, be sure to save the certificate file into your tftp directory and specify your tftp server IP. Both your tftp server and the switch must be on the same subnet.

## 40. Get SSL Key Status

**URL:** /api/get\_ssl\_key\_status

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "ssl_key": {
    "status": "ready in used"
  }
}
```

## 41. Device List Table

**URL:** /api/dev\_list\_table

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "device_list_table":
  [{
    "switch_mac" : "00-11-22-33-44-55",
    "switch_addr" : "192.168.90.3",
    "device_name" : "Switch A",
    "device_list" :
    [{
      "port_no" : 6,
      "poe_used": 0,
      "status": "on",
      "device_type": "SWITCH",
      "model_name": "SM8TAT2SA",
      "device_name": "Switch C",
      "mac": "00-C0-F2-47-A6-FA",
      "ip_addr": "192.168.90.5",
      "rx_rate": 1024,
      "link_partner_port_no": 1,
    }
  ]
}
```

```
    "number_of_alarm_events": 2,
    "events":
    [{
      "date": "2010-01-01",
      "time": "23 51",
      "message": "Higher than maximum throughput limit"
    },
    {
      "date": "2010-01-01",
      "time": "23 52",
      "message": "Higher than maximum throughput limit"
    }
  ],
  {
    "port_no" : 10,
    "poe_used": 0,
    "status": "on",
    "device_type": "SWITCH",
    "model_name": "SM8TAT2SA",
    "device_name": "Switch B",
    "mac": "00-C0-F2-47-A6-F9",
    "ip_addr": "192.168.90.4",
    "rx_rate": 1024,
    "link_partner_port_no": 2,
    "number_of_alarm_events": 0,
    "events": []
  },
  {
    "port_no" : 26,
    "poe_used": 34,
    "status": "on",
    "device_type": "Camera",
    "model_name": "AXIS Camera",
    "device_name": "Camera A",
    "mac": "00-40-8C-7D-81-9A",
    "ip_addr": "192.168.90.203",
    "rx_rate": 1024,
    "link_partner_port_no": 0,
    "number_of_alarm_events": 0,
```

```

        "events":[]
    }
}

```

**Section:**

| Name     | Data type | Unit     |
|----------|-----------|----------|
| poe_used | Integer   | 0.1 watt |
| rx_rate  | Integer   | byte     |

## 42. Get Config File List

**URL:** /api/get\_config\_file\_list

**Method:** GET

**Request JSON:** null

**Response JSON:**

```

{
  "system": {
    "config": {
      "file_list" : [
        {"name":"running-config"},
        {"name":"startup-config"},
        {"name":"default-config"},
        {"name":"test-config"}
      ]
    }
  }
}

```

### 43. Export Config

URL: /api/export\_config

Method: POST

Request JSON:

```
{
  "system": {
    "config": {
      "export_url": "http://192.168.111.1/ ",
      "export_file" : "running-config"
    }
  }
}
```

```
{
  "system": {
    "config": {
      "export_url": "tftp://192.168.111.1/ ",
      "export_file" : "test-config"
    }
  }
}
```

Response JSON: null (note: get action status used "[Get Config Action Status](#)")

Section:

| Name        | Data type | Allowed / Value | Default Value | Note   |
|-------------|-----------|-----------------|---------------|--|
| export_url  | String    | <URL>           |               | Support protocol :<br><b>tftp 、 http</b>   |
| export_file | String    | <File Name>     |               | Special Filename:<br><b>"default-config"</b><br><b>"running-config"</b><br><b>"startup-config"</b> |

## 44. Import Config

**URL:** /api/import\_config

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "config": {
      "import_url": "http://192.168.111.1/test-config",
      "import_file": "running-config",
      "params": "Replace"
    }
  }
}
```

```
{
  "system": {
    "config": {
      "import_url": "tftp://192.168.111.1/test-config",
      "import_file": "test-config"
    }
  }
}
```

**Response JSON:** null (note: get action status used "[Get Config Action Status](#)")

**Section:**

| Name        | Data type | Allowed / Value     | Default Value | Note   |
|-------------|-----------|---------------------|---------------|--|
| import_url  | String    | <URL>               |               | Support protocol :<br><b>tftp</b> 、 <b>http</b>  |
| import_file | String    | <File Name>         |               | Special Filename:<br><b>"default-config"</b> (not supported)<br><b>"running-config"</b><br><b>"startup-config"</b> |
| params      | String    | "Replace" 、 "Merge" | Replace       | <b>"import_file"</b> if <b>"running-config"</b> used.  |



## 45. Activate Config

**URL:** /api/activate\_config

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "config": {
      "activate_file" : "test-config"
    }
  }
}
```

**Response JSON:** null (note: get action status used "[Get Config Action Status](#)")

**Section:**

| Name          | Data type | Allowed / Value | Default Value | Note  |
|---------------|-----------|-----------------|---------------|---|
| activate_file | String    | <File Name>     |               | Special Filename:<br>"default-config" (not supported)<br>"running-config" (not supported)<br>"startup-config" |

## 46. Delete Config

**URL:** /api/delete\_config

**Method:** POST

**Request JSON:**

```
{
  "system": {
    "config": {
      "delete_file" : "test-config"
    }
  }
}
```

**Response JSON:**

```
{
  "system": {
    "config": {
      "delete_status" : "The device has been activate config successfully."
    }
  }
}
```

**Section:**

| Name          | Data type | Allowed / Value                                 | Note   |
|---------------|-----------|---|--|
| delete_file   | String    | <File Name>                                     | Special Filename:<br><b>"default-config"</b> (not supported)<br><b>"running-config"</b> (not supported)<br><b>"startup-config"</b> (not supported) |
| delete_status |           | The device has been delete config successfully. |  |
|               |           | Error: Failed to delete config file.            |  |

## 47. Get Config Action Status

**URL:** /api/get\_config\_action\_status

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "system": {
    "config": {
      "config_file_status": "The device has been import config successfully."
    }
  }
}
```

**Section:**

| Name                      | Data type | Allowed / Value                                   |
|---------------------------|-----------|---|
| <b>config_file_status</b> | String    | Never updated                                     |
|                           |           | Downloading, please stand by...                   |
|                           |           | Processing, please stand by...                    |
|                           |           | The device has been import config successfully.   |
|                           |           | Error: Failed to import config file.              |
|                           |           | The device has been export config successfully.   |
|                           |           | Error: Failed to export config file.              |
|                           |           | The device has been activate config successfully. |
|                           |           | Error: Failed to activate config file.            |

## 48. Ping

**URL:** /api/ping

**Method:** POST

**Request JSON:**

```
{
  "ping": {
    "host" : "192.168.1.1",
    "version" : 4,
    "count" : 5,
    "length" : 56,
    "vlan" : 1
  }
}
```

**Response JSON:**

```
{
  "ping": {
    "status" : "start",
    "message" : ""
  }
}
```

**Note:** get action status used "[Get Ping Status](#)".

**Section:**

| Name    | Data type | Allowed / Value   | Default Value |
|---------|-----------|---|---------------|
| host    | String    | <IPv4 Address><br><IPv6 Address><br><Host Name>   |               |
| version | Integer   | 4 : "host" is <IPv4 Address><br>6 : "host" is <IPv6 Address><br>0 : "host" is <Host Name> | 4             |
| count   | Integer   | 1-60 Times  | 5             |
| length  | Integer   | 1-1452 Bytes  | 56            |
| vlan    | Integer   | <Vlan ID><br>(Note : "host" is <IPv6 Address> used.)                                      | 1             |

## 49. Get Ping Status

**URL:** /api/get\_ping\_status

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "ping": {
    "status" : "done",
    "message" : "PING 192.168.1.1 (192.168.1.1): 56 data bytes\n
      64 bytes from 192.168.1.1: seq=0 ttl=128 time<1ms\n
      64 bytes from 192.168.1.1: seq=1 ttl=128 time<1ms\n
      64 bytes from 192.168.1.1: seq=2 ttl=128 time<1ms\n
      64 bytes from 192.168.1.1: seq=3 ttl=128 time<1ms\n
      64 bytes from 192.168.1.1: seq=4 ttl=128 time=10ms\n\n
      --- 192.168.0.46 ping statistics ---\n
      5 packets transmitted, 5 packets received, 0% packet loss\n
      round-trip min/avg/max = 0.0/2.0/10.0 ms\n"
  }
}
```

**Section:**

| Name | Data type | Allowed / Value                           |
|------|-----------|---|
| stat | String    | "start" 、 "processing" 、 "done" 、 "error" |

## 50. Traceroute

**URL:** /api/traceroute

**Method:** POST

**Request JSON:**

```
{
  "traceroute": {
    "host" : "192.168.1.1",
    "version" : 4,
    "ip_protocol" : "ICMP",
    "wait_time" : 5,
    "max_ttl" : 10,
    "count" : 3,
    "vlan" : 1
  }
}
```

**Response JSON:**

```
{
  "traceroute": {
    "status" : "start",
    "message" : ""
  }
}
```

(note: get action status used "[Get Traceroute Status](#)")

**Section:**

| Name        | Data type | Allowed / Value   | Default Value |
|-------------|-----------|---|---------------|
| host        | String    | <IPv4 Address> <IPv6 Address><br><Host Name>  |               |
| version     | Integer   | 4 : "host" is <IPv4 Address><br>6 : "host" is <IPv6 Address><br>0 : "host" is <Host Name> | 4             |
| ip_protocol | String    | "ICMP" 、 "UDP"  | ICMP          |
| wait_time   | Integer   | 1-60 seconds  | 5             |
| max_ttl     | Integer   | 1-255 hops  | 30            |
| count       | Integer   | 1-10 packets  | 3             |
| vlan        | Integer   | <Vlan ID> (Note : "host" is <IPv6 Address> used.)   | 1             |

## 51 Get Traceroute Status

**URL:** /api/get\_traceroute\_status

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "traceroute": {
    "status" : "done",
    "message" : "traceroute to 192.168.1.1 (192.168.1.1), 30 hops max, 38 byte packets\n
      1 * * *\n
      2 * * *\n
      3 * * *\n
      4 * * *\n
      5 * * *\n
      6 * * *\n
      7 * * *\n
      8 * * *\n
      9 * * *\n
      10 * * * \n"
  }
}
```

**Section:**

| Name | Data type | Allowed / Value                           |
|------|-----------|---|
| stat | String    | "start" 、 "processing" 、 "done" 、 "error" |

## 52. Get SNMP Trap Config

**URL:** /api/get\_snmp\_trap\_config

**Method:** GET

**Request JSON:** null

**Response JSON:**

```
{
  "snmp":{
    "trap": [{
      "id": 2,
      "mode": "UDP",
      "version": "SNMPv2c",
      "community": "public",
      "dest_addr": "192.168.1.1",
      "dest_port": 162,
      "severity_level": "Error"
    },
    ... ..
  ]
}
```



## 53. Add SNMP Trap Config

**URL:** /api/add\_snmp\_trap\_config

**Method:** POST

**Request JSON:**

```
{
  "snmp": {
    "trap": {
      "add": [{
        "id": 2,
        "mode": "udp",
        "version": "SNMPv2c",
        "community": "public",
        "dest_addr": "192.168.1.1",
        "dest_port": 162,
        "severity_level": "Error"
      },
      ... ..
    ]
  }
}
```

**Response JSON:**

```
{
  "snmp": {
    "trap": [{
      "id": 2,
      "mode": "UDP",
      "version": "SNMPv2c",
      "community": "public",
      "dest_addr": "192.168.1.1",
      "dest_port": 162,
      "severity_level": "Error"
    },
    ... ..
  ]
}
```

## Section:

| Name           | Data type | Allowed / Value   | Default Value |
|----------------|-----------|---|---------------|
| id             | Integer   | <Entry Number 1-6>  |               |
| mode           | String    | "Disabled" 、 "UDP" 、 "TCP"  | Disabled      |
| version        | String    | "SNMPv2c"   | SNMPv2c       |
| community      | String    | 0-32 alphanumeric   |               |
| dest_addr      | String    | <IPv4 Address>  |               |
| dest_port      | Integer   | 1-65535   | 162           |
| severity_level | String    | "Emerg" 、 "Alert" 、 "Crit" 、 "Error"<br>"Warning" 、 "Notice" 、 "Info" 、 "Debug" | Emerg         |

## 54. Delete SNMP Trap Config

**URL:** /api/del\_snmp\_trap\_config

**Method:** POST

**Request JSON:**

```
{
  "snmp": {
    "trap": {
      "delete": [{
        "id": 5
      },
      ... ..
    ]
  }
}
```

**Response JSON:**

```
{
  "snmp": {
    "trap": [ {
      "id": 2,
      "mode": "UDP",
      "version": "SNMPv2c",
      "community": "public",
      "dest_addr": "192.168.1.1",
      "dest_port": 162,
      "severity_level": "Error"
    },
    ... ..
  ]
}
```

**Section:**

| Name | Data type | Allowed / Value    | Default Value |
|------|-----------|--------------------|---------------|
| id   | Integer   | <Entry Number 1-6> |               |

## cURL Commands v 1.7

```

curl -v --cookie "seid=123456789" -d '{"login":{"username":"admin","password":"admin","user_ip":"192.168.1.77","sessid":"123456789"}}' http://192.168.1.77/api/login

curl -v --cookie "seid=123456789" -d '{"logout":{"sessid":"123456789"}}'
http://192.168.1.77/api/logout

curl -v --cookie "seid=123456789" -d '{"system":{"warm":"Yes"}}' http://192.168.1.77/api/reboot

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_sysinfo

curl -v --cookie "seid=123456789" -d '{"system":{"information":{"system_name":"SM16TAT2SA"},"location":"Minnetonka","contact":"Tech supportt"}}'
http://192.168.1.77/api/set_sysinfo

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_poe_status

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_poe_config

curl -v --cookie "seid=123456789" -d
 '{"poe":{"power_determined_mode":"Class","capacitor_detection":false,"ports":[{"id":1,"poe
":{"Mode":"Enabled","Priority":"Low","power_limit_user":30,"schedule":"Disabled"}}}'
http://192.168.1.77/api/set_poe_config

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_poe_auto_reset

curl -v --cookie "seid=123456789" -d '{"poe":{"auto_checking":false,"ports":[{"id":
1,"poe_auto_reset":{"ip":"192.168.1.1","startup_time":60,"interval_time":30,"retry_time":
3,"failure_reboot":false,"reboot_time":15,"max_reboot_times":3}}}'
http://192.168.1.77/api/set_poe_auto_reset

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_port_statistics

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_port_config

curl -v --cookie "seid=123456789" -d '{"ports":[{"id":1,"speed_mode":"Auto","flow_control":
false,"description":"test"}]}' http://192.168.1.77/api/set_port_config

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_sfp_port_detail

curl -v --cookie "seid=123456789" -d '{"system":{"firmware":
{"upgrade_url":"http://192.168.5.46/test.tar.gz"}}}' http://192.168.1.77/api/firmware_upgrade

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_firmware_upgrade_status

curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_account_config

curl -v --cookie "seid=123456789" -d '{"account":{"status":"NEW","username":
"superuser","password":"superuser","privilege_level":15}}'

```

`http://192.168.1.77/api/set_account_config`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_dynamic_mac_table`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/save_configuration`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_system_time`

`curl -v --cookie "seid=123456789" -d '{"system\":{"time\":{"clock_source\":"Local Setting\","system_date\":"2020-07-01 01:01:30\","time_zone\":"5400\","acronym\":"\","daylight\":{"mode\":"false","offset\":"60","start_time\":{"month\":"Jan\","week\":"1","day\":"Mon\","hour\":"1"},"end_time\":{"month\":"Jan\","week\":"1","day\":"Mon\","hour\":"1"}}}}}' http://192.168.1.77/api/set_system_time`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_ntp_server`

`curl -v --cookie "seid=123456789" -d '{"system\":{"ntp\":{"server\":[{"id\":"1","host\":"192.168.1.1"},"interval\":"60"]}}}' http://192.168.1.77/api/set_ntp_server`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_syslog_server`

`curl -v --cookie "seid=123456789" -d '{"event_notification\":{"syslog\":{"mode\":"false","server\":[{"id\":"1","address\":"192.168.1.1"}]}}}' http://192.168.1.77/api/set_syslog_server`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_syslog`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/clear_syslog`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_vlan_config`

`curl -v --cookie "seid=123456789" -d '{"vlan\":{"allowed_access_vlans\":"1\","ethertype_custom_s_ports\":"88a8"},"ports\":[{"id\":"2","vlan\":{"mode\":"Access\","access\":{"pvid\":"1"}},{ "id\":"3","vlan\":{"mode\":"Trunk\","trunk\":{"pvid\":"1","egress_tagging\":"Untag Port VLAN\","allowed_vlan\":"1"}},{ "id\":"4","vlan\":{"mode\":"Hybrid\","hybrid\":{"pvid\":"1","port_type\":"C-Port\","ingress_filter\":"false","ingress_accept\":"Tagged and Untagged\","egress_tagging\":"Untag Port VLAN\","allowed_vlan\":"1"}},]}`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_mac_based_vlan`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_ip_address`

`curl -v --cookie "seid=123456789" -d '{"system\":{"ip\":{"interfaces\":[{"vid\":"1","ipv4\":{"dhcp\":"false","fallback\":"0","static_addr\":"192.168.111.126","static_mask\":"24"},"ipv6\":{"static_addr\":"\","static_mask\":"0"}]}}}' http://192.168.1.77/api/set_ip_address`

`curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_mirror_config`

`curl -v --cookie "seid=123456789" -d '{"system\":{"mirror\":[{"session\":"1","enable\":"false","destination_port\":"2","source_tx\":"4,6-8","source_rx\":"3,5,7-8"}]}}}'`

```
http://192.168.1.77/api/set_mirror_config
```

```
curl -v --cookie "seid=123456789" -d '{"cable": {"port": 5}}' http://192.168.1.77/api/cable_diagnostics
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_acl_config
```

```
curl -v --cookie "seid=123456789" -d '{"acl": {"ace_id": 3, "next_ace_id": 0, "ingress_port_list":  
"1", "frame_type": "Any", "action": "Deny", "mirror": true, "counter": true, "metering":  
{"enable": true, "bandwidth": 900000}}}' http://192.168.1.77/api/set_acl_config
```

```
curl -v --cookie "seid=123456789" -d '{"ssl_key":  
{"create": {"country": "US", "state": "MN", "location": "Minnetonka", "organization": "Transition  
Networks", "organizational_unit": "IES", "common_name": "IES", "email": "techsupport@transition  
.com", "days": 3650, "key_length": 2048}}}' http://192.168.1.77/api/create_ssl_key
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_ssl_key_status
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/dev_list_table
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_config_file_list
```

```
curl -v --cookie "seid=123456789" -d '{"system": {"config": {"export_url": "http://192.168.111.1/  
", "export_file": "running-config"}}}' http://192.168.1.77/api/export_config
```

```
curl -v --cookie "seid=123456789" -d '{"system": {"config": {"import_url":  
"http://192.168.111.1/test-config", "import_file": "running-config", "params": "Replace"}}}'  
http://192.168.1.77/api/import_config
```

```
curl -v --cookie "seid=123456789" -d '{"system": {"config": {"activate_file": "test-config"}}}'  
http://192.168.1.77/api/activate_config
```

```
curl -v --cookie "seid=123456789" -d '{"system": {"config": {"delete_file": "test-config"}}}'  
http://192.168.1.77/api/delete_config
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_config_action_status
```

```
curl -v --cookie "seid=123456789" -d '{"ping": {"host": "192.168.1.1", "version": 4, "count":  
5, "length": 56, "vlan": 1}}' http://192.168.1.77/api/ping
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_ping_status
```

```
curl -v --cookie "seid=123456789" -d '{"traceroute": {"host": "192.168.1.1", "version":  
4, "ip_protocol": "ICMP", "wait_time": 5, "max_ttl": 5, "count": 3, "vlan": 1}}'  
http://192.168.1.77/api/traceroute
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_traceroute_status
```

```
curl -v --cookie "seid=123456789" http://192.168.1.77/api/get_snmp_trap_config
```

```
curl -v --cookie "seid=123456789" -d '{"snmp": {"trap_mode": false, "trap": {"add": [{"id":  
2, "mode": "UDP", "version": "SNMPv2c", "community": "public", "dest_addr":
```

```
\\"123.123.123.123\\",\\"dest_port\\": 55,\\"severity_level\\": \\"Error\\"}]]]]"
http://192.168.1.77/api/add_snmp_trap_config
```

```
curl -v --cookie "seid=123456789" -d "{\\"snmp\\": {\\"trap_mode\\": false,\\"trap\\": {\\"delete\\": [{\\"id\\":
5}}]]}" http://192.168.1.77/api/del_snmp_trap_config
```

## Record of Revisions

| Rev. | Date    | Description  |
|------|---------|--|
| A    | 8/11/20 | Initial release for SMxTAT2SA Series at FW v1.02.1476, API-04172020 (003) and SM8TAT2SA-DC FW v1.02.1476.  |
| B    | 8/9/21  | FW v1.04.0020: update Device List Table. Change power_determined_mode default to "Class" on page 16. Fix API file updating and fix API issues with ACLs. Add API commands for Ping and Traceroute. Add "copy startup-config to running-config" in Replace mode. Add download/upload config file and Show Power Override. Add "PoE profile list" on get PoE config and FW upgrade status messages. Add "import / export config file" and "activate / delete config file". Fix get PoE status data. Fix LLDP neighbor management address issue and PoE Mode field issue after IP camera reboot. Add cURL Commands v 1.7. |

**Note:** Minimum version of firmware required: v7.10.1476.

### Additions:

|   |   |
|---|---|
| <p><b><u>FW v1.04.0013:</u></b></p> <ol style="list-style-type: none"> <li>1. Get Config File List</li> <li>2. Export Config</li> <li>3. Import Config</li> <li>4. Activate Config</li> <li>5. Delete Config</li> <li>6. Get Config Action Status</li> <li>7. Ping</li> <li>8. Get Ping Status</li> <li>9. Traceroute</li> <li>10. Get Traceroute Status</li> </ol> | <p><b><u>FW v 1.04.0018:</u></b></p> <ol style="list-style-type: none"> <li>1. Get PoE Auto Power Reset</li> <li>2. Get PoE Auto Power Reset</li> <li>3. Get SFP Port Detail</li> <li>4. Get Syslog Log</li> <li>5. Clear Syslog</li> <li>6. Get SNMP Trap Config</li> <li>7. Add SNMP Trap Config</li> <li>8. Delete SNMP Trap Config</li> </ol> |
|---|---|