

API Reference

# flexiEdge UI API

API Version: 1.7.3

flexiEdge UI API. Find more at <https://docs.flexiwan.com>.

## CONTACT

EMAIL: [yourfriends@flexiwan.com](mailto:yourfriends@flexiwan.com)

Terms of service: <https://flexiwan.com/legal/terms-of-service/>

# INDEX

<b>1. DEVICE</b>	<b>4</b>
1.1 GET /device	4
1.2 PUT /device	5
1.3 GET /interfaces	5
1.4 PUT /interfaces	6
1.5 PUT /uuid	7
1.6 GET /token	8
1.7 PUT /token	8
1.8 GET /dump	9
1.9 GET /connectivity	9
<b>2. INTERFACE</b>	<b>11</b>
2.1 GET /interfaces	11
2.2 PUT /interfaces	12
<b>3. SYSTEM</b>	<b>14</b>
3.1 GET /systemcheckers	14
3.2 PUT /systemcheckers	14
3.3 PUT /systemcheckers/{systemcheckername}	15
3.4 GET /logs	15
<b>4. USER</b>	<b>17</b>
4.1 PUT /login	17
4.2 POST /login	17

# Security and Authentication

## SECURITY SCHEMES

KEY	TYPE	DESCRIPTION
BearerAuth	http, bearer, JWT	

# API

## 1. DEVICE

Everything about device

### 1.1 GET /device

#### Device information

Gets device information, such as version, id, serial number, connection status etc.

#### REQUEST

No request parameters

#### RESPONSE

STATUS CODE - 200: successful operation

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
  deviceManagement  enum    ALLOWED:running, stopped  
                        device management status  
  deviceSerialNumber string  device serial number. When not detected (i.e virtual machine), set to N/A  
  hostname           string  device hostname  
  machineId          string  machine id  
  vRouter {  
    state  enum    ALLOWED:running, stopped  
            router state  
    error  string  router error information  
  }  
  versions {  
    components {  
      agent {  
        version string  component version  
      }  
      edgeui {  
        version string  component version  
      }  
      frr {  
        version string  component version  
      }  
      router {  
        version string  component version  
      }  
      vpp {  
        version string  component version  
      }  
    }  
  }  
  device  string  device version  
}]
```

STATUS CODE - 400: Invalid status value

## 1.2 PUT /device

### Apply device command

Applies device command

#### REQUEST

REQUEST BODY - application/json

```
{
  command* enum ALLOWED:stopRouter, startDeviceManagement, stopDeviceManagement,
              clearDeviceConfiguration, reset, completeReset, reboot
              device command to be executed
}
```

#### RESPONSE

STATUS CODE - 200: Command executed successfully

RESPONSE MODEL - application/json

```
[{
  Array of object:
    status      string  execution status
    description  string  execution description
}]
```

STATUS CODE - 500: failed operation

RESPONSE MODEL - application/json

```
[{
  Array of object:
    error string  error description
}]
```

## 1.3 GET /interfaces

### Get device interfaces

Gets the available device interfaces

#### REQUEST

No request parameters

#### RESPONSE

STATUS CODE - 200: successful operation

RESPONSE MODEL - application/json

```
[{
  Array of object:
    name          string  interface name
    IPv4          string  IPv4 address
                  string  IPv4 address mask
}
```

```

IPv4Mask      string
MAC           string      MAC address
dhcp         enum        ALLOWED:yes, no
                        address allocation status
gateway      string      default gateway address
internetAccess enum      ALLOWED:yes, no
                        is interface connected to internet
metric       string      interface metric
link         enum        ALLOWED:up, down
                        interface link status

id           integer

pppoe {
  config {
    is_enabled boolean  enables or disables PPPoE connection
    metric     string    interface metric
    mru        string    maximum reception unit
    mtu        string    maximum transmission unit
    password   string    PPPoE connection password
    username   string    PPPoE connection user name
    usepeerdns boolean   when true, asks for the nameservers from peer
    nameservers [string]
  }
  status {
    addr      string    interface IP address and mask
    gw        string    interface default gateway
    is_connected boolean indicates the status of PPPoE connection
  }
}
}
}]

```

STATUS CODE - 500: Internal Server Error

## 1.4 PUT /interfaces

### Update interface

Updates interface configuration

### REQUEST

REQUEST BODY - application/json

```

{
  name          string  interface name
  IPv4          string  IPv4 address
  IPv4Mask      string  IPv4 address mask
  MAC           string  MAC address
  dhcp         enum    ALLOWED:yes, no
                        address allocation status
  gateway      string  default gateway address
  internetAccess enum  ALLOWED:yes, no
                        is interface connected to internet
  metric       string  interface metric. When empty or omit, will be set to "0"
  link         enum    ALLOWED:up, down
                        interface link status

  pppoe {
    is_enabled boolean  enables or disables PPPoE connection
    metric     string    interface metric
  }
}

```

```

    mru          string    maximum reception unit
    mtu          string    maximum transmission unit
    password     string    PPPoE connection password
    username     string    PPPoE connection user name
    usepeerdns   boolean   when true, asks for the nameservers from peer
    nameservers  [string]
  }
}

```

## RESPONSE

**STATUS CODE - 200:** Interface update executed successfully

### RESPONSE MODEL - application/json

```

{
  name          string    interface name
  IPv4          string    IPv4 address
  IPv4Mask      string    IPv4 address mask
  MAC           string    MAC address
  dhcp          enum      ALLOWED:yes, no
                  address allocation status
  gateway       string    default gateway address
  internetAccess enum    ALLOWED:yes, no
                  is interface connected to internet
  metric        string    interface metric
  link          enum      ALLOWED:up, down
                  interface link status
  id            integer
  pppoe {
    config {
      is_enabled boolean  enables or disables PPPoE connection
      metric     string    interface metric
      mru        string    maximum reception unit
      mtu        string    maximum transmission unit
      password   string    PPPoE connection password
      username   string    PPPoE connection user name
      usepeerdns boolean   when true, asks for the nameservers from peer
      nameservers [string]
    }
    status {
      addr       string    interface IP address and mask
      gw         string    interface default gateway
      is_connected boolean  indicates the status of PPPoE connection
    }
  }
}

```

**STATUS CODE - 500:** failed operation

### RESPONSE MODEL - application/json

```

[ {
  error string  error description
} ]

```

## 1.5 PUT /uuid

## Sets the new machine ID

Sets the new device/machine ID

### REQUEST

```
REQUEST BODY - application/json
{
  machineId string machine ID
}
```

### RESPONSE

STATUS CODE - 200: Updated Uuid

```
RESPONSE MODEL - application/json
string
```

## 1.6 GET /token

### Get Device Token

Gets device token. Device token is needed in order to establish connectivity to flexiManage.

### REQUEST

No request parameters

### RESPONSE

STATUS CODE - 200: successful operation

RESPONSE MODEL - application/json

```
[{
  Array of object:
  deviceToken string device token
}]
```

STATUS CODE - 500: failed operation

RESPONSE MODEL - application/json

```
[{
  Array of object:
  error string error description
}]
```

## 1.7 PUT /token

### Update device token

Updates device token. Device token is needed in order to establish connectivity to flexiManage.

### REQUEST

REQUEST BODY - application/json

```
{
  deviceToken string device token
}
```

## RESPONSE

**STATUS CODE - 200:** successful operation

**RESPONSE MODEL - application/json**

```
[{
  Array of object:
    deviceToken string device token
}]
```

**STATUS CODE - 500:** failed operation

**RESPONSE MODEL - application/json**

```
[{
  Array of object:
    error string error description
}]
```

## 1.8 GET /dump

### Fwdump

Gets fwdump file

## REQUEST

No request parameters

## RESPONSE

**STATUS CODE - 200:** successful operation

**STATUS CODE - 500:** failed operation

**RESPONSE MODEL - application/json**

```
[{
  Array of object:
    error string error description
}]
```

## 1.9 GET /connectivity

### Connectivity checks

Performs device connectivity checks by using long polling

## REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
------	------	-------------

NAME	TYPE	DESCRIPTION
*type	enum ALLOWED: continuousping, ping, traceroute	The type of check to perform
*server	string	The address of the server

## RESPONSE

**STATUS CODE - 200: successful operation**

**RESPONSE MODEL - text/event-stream**

```
[{
  Array of object: Login response (Success case)
  access_token string Bearer Authorization Token (JWT)
  token_type string Bearer Authorization Token (JWT) type
  name string Logged in user name
  status string Login status
}]
```

**STATUS CODE - 500: failed operation**

**RESPONSE MODEL - application/json**

```
[{
  Array of object:
  error string error description
}]
```

## 2. INTERFACE

Device interfaces information and update

### 2.1 GET /interfaces

#### Get device interfaces

Gets the available device interfaces

#### REQUEST

No request parameters

#### RESPONSE

**STATUS CODE - 200:** successful operation

**RESPONSE MODEL - application/json**

```
[{  
  Array of object:  
  name          string      interface name  
  IPv4          string      IPv4 address  
  IPv4Mask      string      IPv4 address mask  
  MAC           string      MAC address  
  dhcp          enum        ALLOWED:yes, no  
                                address allocation status  
  gateway       string      default gateway address  
  internetAccess enum        ALLOWED:yes, no  
                                is interface connected to internet  
  metric        string      interface metric  
  link          enum        ALLOWED:up, down  
                                interface link status  
  id            integer  
  pppoe {  
    config {  
      is_enabled boolean    enables or disables PPPoE connection  
      metric      string      interface metric  
      mru         string      maximum reception unit  
      mtu         string      maximum transmission unit  
      password    string      PPPoE connection password  
      username    string      PPPoE connection user name  
      usepeerdns  boolean    when true, asks for the nameservers from peer  
      nameservers [string]  
    }  
    status {  
      addr        string      interface IP address and mask  
      gw          string      interface default gateway  
      is_connected boolean    indicates the status of PPPoE connection  
    }  
  }  
}]
```

**STATUS CODE - 500:** Internal Server Error

## 2.2 PUT /interfaces

### Update interface

Updates interface configuration

#### REQUEST

REQUEST BODY - application/json

```
{
  name          string  interface name
  IPv4          string  IPv4 address
  IPv4Mask      string  IPv4 address mask
  MAC           string  MAC address
  dhcp          enum    ALLOWED:yes, no
                  address allocation status
  gateway       string  default gateway address
  internetAccess enum  ALLOWED:yes, no
                  is interface connected to internet
  metric        string  interface metric. When empty or omit, will be set to "0"
  link          enum    ALLOWED:up, down
                  interface link status

  pppoe {
    is_enabled  boolean  enables or disables PPPoE connection
    metric      string   interface metric
    mru         string   maximum reception unit
    mtu         string   maximum transmission unit
    password    string   PPPoE connection password
    username    string   PPPoE connection user name
    usepeerdns  boolean  when true, asks for the nameservers from peer
    nameservers [string]
  }
}
```

#### RESPONSE

STATUS CODE - 200: Interface update executed successfully

RESPONSE MODEL - application/json

```
{
  name          string  interface name
  IPv4          string  IPv4 address
  IPv4Mask      string  IPv4 address mask
  MAC           string  MAC address
  dhcp          enum    ALLOWED:yes, no
                  address allocation status
  gateway       string  default gateway address
  internetAccess enum  ALLOWED:yes, no
                  is interface connected to internet
  metric        string  interface metric
  link          enum    ALLOWED:up, down
                  interface link status

  id            integer

  pppoe {
    config {
      is_enabled  boolean  enables or disables PPPoE connection
      metric      string   interface metric
    }
  }
}
```

```
    mru          string    maximum reception unit
    mtu          string    maximum transmission unit
    password     string    PPPoE connection password
    username     string    PPPoE connection user name
    usepeerdns   boolean   when true, asks for the nameservers from peer
    nameservers  [string]
  }
  status {
    addr         string    interface IP address and mask
    gw          string    interface default gateway
    is_connected boolean   indicates the status of PPPoE connection
  }
}
}
```

**STATUS CODE - 500: failed operation**

**RESPONSE MODEL - application/json**

```
[{
  Array of object:
  error string error description
}]
```

---

## 3. SYSTEM

System related operations

### 3.1 GET /systemcheckers

#### System checkers

Gets the list of available system checkers

#### REQUEST

No request parameters

#### RESPONSE

**STATUS CODE - 200:** successful operation

**RESPONSE MODEL - application/json**

```
[{  
  Array of object:  
  name      string  system checker name (id)  
  description string  system checker description set to N/A  
  category   enum    ALLOWED:hard, soft  
              system checker category  
  severity   enum    ALLOWED:critical, optional  
              system checker severity  
  result     enum    ALLOWED:Passed, Failed, Skipped  
              system checker run result  
}]
```

### 3.2 PUT /systemcheckers

#### Run all system checkers

Runs all system checkers at once

#### REQUEST

**REQUEST BODY - application/json**

```
{  
  autoFix boolean when set to true, tries to silently fix failed checks  
}
```

#### RESPONSE

**STATUS CODE - 200:** successful operation

**RESPONSE MODEL - application/json**

```
[{  
  Array of object:  
  name      string  system checker name (id)  
  description string  system checker description set to N/A
```

```

    category    enum    ALLOWED:hard, soft
                system checker category
    severity    enum    ALLOWED:critical, optional
                system checker severity
    result      enum    ALLOWED:Passed, Failed, Skipped
                system checker run result
  }}

```

### 3.3 PUT /systemcheckers/{systemcheckername}

#### Run specific system checker

Runs specified system checker

#### REQUEST

##### PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*systemcheckername	string	System checker name to run

##### REQUEST BODY - application/json

```

{
  autoFix boolean when set to true, tries to silently fix failed checks
}

```

#### RESPONSE

STATUS CODE - 200: successful operation

##### RESPONSE MODEL - application/json

```

[ {
  Array of object:
    name      string  system checker name (id)
    description string  system checker description set to N/A
    category  enum    ALLOWED:hard, soft
                system checker category
    severity  enum    ALLOWED:critical, optional
                system checker severity
    result    enum    ALLOWED:Passed, Failed, Skipped
                system checker run result
  } ]

```

STATUS CODE - 500: failed operation

##### RESPONSE MODEL - application/json

```

[ {
  Array of object:
    error string error description
  } ]

```

### 3.4 GET /logs

#### System logs

Gets various system logs

## REQUEST

### QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
limit	integer	The numbers of items to return
filter	enum ALLOWED: fwagent, syslog, dhcp, vpp, ospf, hostapd	Filter to be applied

## RESPONSE

STATUS CODE - 200: successful operation

### RESPONSE MODEL - application/json

```
[{  
  Array of object:  
  name      string  system checker name (id)  
  description string  system checker description set to N/A  
  category  enum    ALLOWED:hard, soft  
              system checker category  
  severity  enum    ALLOWED:critical, optional  
              system checker severity  
  result    enum    ALLOWED:Passed, Failed, Skipped  
              system checker run result  
}]
```

STATUS CODE - 500: failed operation

### RESPONSE MODEL - application/json

```
[{  
  Array of object:  
  error string  error description  
}]
```

## 4. USER

Everything about user

### 4.1 PUT /login

#### Modify user password

Modifies user password

#### REQUEST

##### REQUEST BODY - application/json

```
[{  
  Array of object: User credentials. Passwords are base64 encoded  
  username    string  user name  
  password    string  base64 encoded current user password  
  newPassword string  base64 encoded new user password  
}]
```

#### RESPONSE

STATUS CODE - 200: successful operation

##### RESPONSE MODEL - application/json

```
[{  
  Array of object:  
  name    string  user name  
  status  string  operation status  
}]
```

### 4.2 POST /login

#### System login

Performs system login. By successfully logging in, the user will be granted an Authorization token

#### REQUEST

##### REQUEST BODY - application/json

```
[{  
  Array of object: User credentials. Passwords are base64 encoded  
  username string  user name  
  password string  base64 encoded current user password  
}]
```

#### RESPONSE

STATUS CODE - 200: successful operation

##### RESPONSE MODEL - application/json

```
[{  
  Array of object: Login response (Success case)
```

```
access_token string Bearer Authorization Token (JWT)
token_type string Bearer Authorization Token (JWT) type
name string Logged in user name
status string Login status
}]
```

**STATUS CODE - 401: unsuccessful operation**

**RESPONSE MODEL - application/json**

```
[{
  Array of object: Login response (Failure case)
  error string Error description
}]
```

---