

KIWIRE 2.0

API2 DOCUMENTATION

VERSION 2.0 (DECEMBER 2017)

Proprietary Information Notice

This document is proprietary to Synchronweb (M) Sdn. Bhd. By utilizing this document, the recipient agrees to avoid publication or any other unrestricted disclosure of any part of this document or the information it contains and to only make copies needed for recipient's internal review.

Implementation Notes

- 1) The Licensee, e.g. Client of Synchronweb (M) Sdn. Bhd., including this API, has the right to use or subcontract the use of this API for work related to their implementation and data.
- 2) The Licensee will receive a new API account, e.g. authentication key, upon request. Each subcontractor shall use a unique account in all their usage of this API.
- 3) Each and all distinct projects using this API must be approved and quality assured by Synchronweb prior to implementation and launch, including projects developed internally by Licensee. Synchronweb will make recommendations to change functionalities that are deemed to adversely affect performance and security of any and all Synchronweb related system or services. In the case where the API service is running on a server owned and controlled by Synchronweb, such recommendations must be strictly adhered to.
- 4) Synchronweb quality assurance and assistance for each and all API projects is not covered by any prior agreements between Synchronweb and Licensee, and a quotation for such services must be requested prior to implementation. Synchronweb will study project scopes and make an assessment of the work deemed reasonable to assure project quality.
- 5) Synchronweb support to Licensee is restricted to proper working of the functions within this API. Any usage of such functions by Licensee and Subcontractors, where the functions are deemed by Synchronweb to be working in accordance with the specification in this document, is under the sole responsibility and risk by the Licensee. Any loss or damage to Licensee data under such conditions are not the responsibility of Synchronweb, and as such the Licensee shall not hold Synchronweb liable.
- 6) Synchronweb will assist to best ability in case of damage recovery, and Licensee shall compensate Synchronweb for such work by prior agreed daily service rate, or of no such agreements exist, by a rate quoted by Synchronweb when such damage recovery is ordered by Licensee.

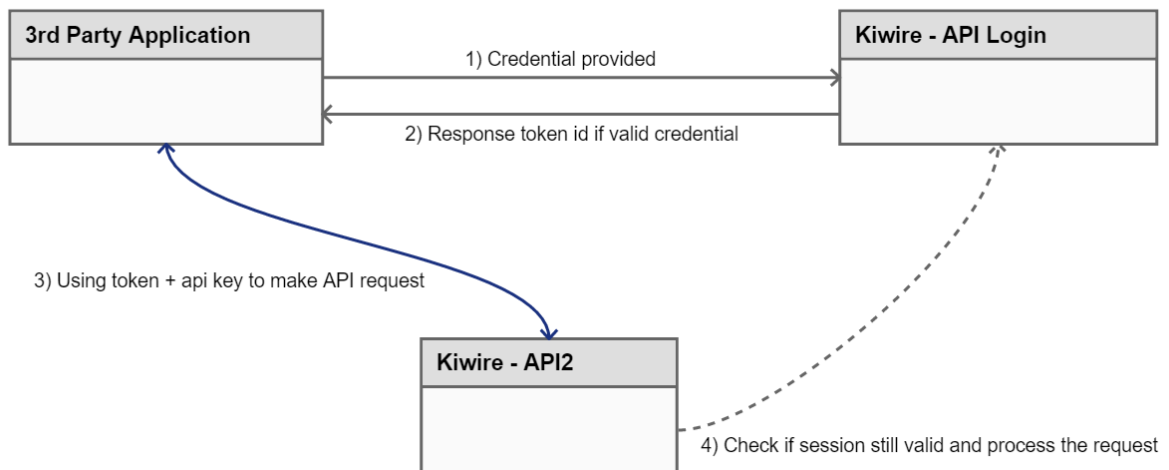
- 7) All API projects shall be documented in scope and functionality by Licensee and Subcontractor before starting any implementation. This documentation shall be submitted to, and approved by Synchronweb as part of quality assurance project ordered in conjunction with the project. Such documentation shall contain:
 - a. Project description and related use cases, e.g. what is the intended usage of this project.
 - b. Technology description, e.g. what programming languages and other technologies will be used in this project. This also includes hardware and computers involved in the operations and specifications of the network connections used by any implemented API clients.
 - c. List of API functions that are planned to be used in the project, together with frequency estimation, e.g. how frequent are the functions expected to be called.
- 8) As part of the quality assurance project, Synchronweb will assist Licensee or Subcontractor in producing the above mentioned documentation, by answering questions and provide knowledge, but not in the actual production of such documents.
- 9) If training is requested, Synchronweb will include such request in the quotation for quality assurance project.

KIWIRE 2.0 – API2

GENERAL PREFACE

This document to explain how 3rd party software can access and modify Kiwire and this document only meant to be use with Kiwire build 8 and above.

REQUEST FLOW



KIWIRE 2.0 – API2

GENERAL REQUEST URL

Kiwire API2 can be trigger using this URL;

[POST] `http://{{kiwire_ip}}/api2/{{module}}/{{method}}/{{value}}`

where;

| | | |
|----------------------------|---|-----------------------------|
| <code>{{kiwire_ip}}</code> | = | Kiwire IP address |
| <code>{{module}}</code> | = | Module |
| <code>{{method}}</code> | = | Create / Delete / View |
| <code>{{value}}</code> | = | Specify item for get method |

Notes;

Currently, Kiwire only provided API for these modules;

- 1) Admin User
- 2) NAS
- 3) Devices
- 4) Profile
- 5) Login
- 6) User
- 7) Voucher

General Result

Every API request send to Kiwire will be respond as a JSON object with the below structure;

- 1) `is_success` – boolean (always true)
- 2) `result_msg` – string (message generated as a result of a request)
- 3) `ret_data` – array (items for get method)

Besides that, Kiwire also will respond error message as per below structure;

- 1) `error` – string (error message)

KIWIRE 2.0 – API2

API LOGIN

Every API request need to perform login to obtain token key which then will be used to make request to module.

URL

`http://{{kiwire_ip}}/admin/agent/api_login.php`

PARAMETERS REQUIRED

| Parameter | Description |
|-----------------------|--|
| <code>code</code> | MD5 hashed of the username password Example: username is hakim and password is admin, then the code will be <code>md5("hakim admin") = 404ad50c254f12a606ae68d0338e86a9</code> |
| <code>cloud_id</code> | Cloud ID or Tenant name of the API key |

RESPONSE

```
{  
  "status": "success",  
  "token": "80d588720c2203739ad643d405e63136"  
}
```

NOTES

This action will create a API session inside Kiwire which expired after 20 minutes. You have to re-login after 20 minutes to continue making API request.

The token provided will be use in all API request.

KIWIRE 2.0 – API2

MODULE: LOGIN

URL

[POST] `http://{{kiwire_ip}}/apiv2/login/{{method}}/{{value}}`

Where `{{method}}` can be;

- history : *List of sessions created for particular account*
- online : *Send request to NAS for login user device as per account provided*
- unauthorize : *List of user device which are not authorised*

Where `{{value}}` can be;

- username : For *history* method
- nas id : For *unauthorize* method

PARAMETERS REQUIRED

History - List of sessions created for particular account

| Parameter | Value | Description |
|-------------------|-------------------|---|
| start_time | date(Y-m-d H:i:s) | (optional) From login datetime |
| end_time | date(Y-m-d H:i:s) | (optional) Until login datetime |
| start | integer | (optional) (<i>default: 0</i>) Start from this row number |
| limit | integer | (optional) (<i>default: 50</i>) Limit data count |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

* If username is not provided in this request, the session list for all user will be returned.

Online - Send request to NAS for login user device as per account provided

| Parameter | Value | Description |
|-----------------|--------------|---|
| nas-id | string | NAS Identity to send the request |
| device | string | MAC Address of the user device |
| ip | string | IP Address of the user device |
| user | string | Username to be used to login user device |
| profile | string | (optional) Profile to link user with |
| allownas | string | (optional) Zone to restrict user |
| status | string | (optional) Status of user (active or suspend) |
| remark | string | (optional) Remark |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

* Parameters marked as optional will only be used if user creation required (user not existed). If these values are not provided during the creation process, default value will be used as per below;

| Parameter | Default Value |
|-----------------|-----------------------------------|
| profile | wx_temp, temporary 5 mins profile |
| allownas | none |
| status | active |
| remark | API Registration |

Unauthorize - List of user device which are not authorised

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

RESPONSE

```
{
  "is_success" : "true", // always true to indicate that the request has been processed
  "result_message" : "", // return a string of status for online method
  "return_data" : [] // return items found in database for unauthorize and history
}
```

KIWIRE 2.0 – API2

MODULE: ADMINISTRATOR USER

URL

[POST] `http://{{kiwire_ip}}/apiv2/admin/{{method}}/{{value}}`

Where `{{method}}` can be;

- create - delete - get

Where `{{value}}` can be;

- administrator username

PARAMETERS REQUIRED

Create

| Parameter | Value | Description |
|------------------|--------------|--|
| username | string | Username to login into Kiwire system |
| password | string | Password to login into Kiwire system |
| groupname | string | Role for the admin |
| fullname | string | Full name |
| monitor | y or n | Set the value to Y if this admin should receive notification email from Kiwire |
| email | string | Email address to send notification |
| temp_pass | boolean | (Optional) (<i>default: true</i>) Set value to true if this admin should change password after login |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Delete

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Get

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |
| start | integer | (optional) (<i>default: 0</i>) Number of start data |
| limit | integer | (optional) (<i>default: 200</i>) Limit data count |

NOTES

For get method, if **{{value}}** not provided then Kiwire will return all available items subject to **start** and **limit**.

KIWIRE 2.0 – API2

MODULE: USER ACCOUNT

URL

[POST] `http://{{kiwire_ip}}/apiv2/user/{{method}}/{{value}}`

Where {{method}} can be;

- create - delete - get

Where {{value}} can be;

- user account

PARAMETERS REQUIRED

Create

| Parameter | Value | Description |
|--------------------|--------|---|
| username | string | Username for wifi authentication |
| password | string | Password for wifi authentication |
| fullname | string | Full name |
| email | string | Email address |
| phone | string | Phone number |
| profile | string | Internet plan |
| remark | string | Remark |
| status | string | |
| allownas | string | |
| expiry | string | The validity of account to be generated |
| mac | string | Mac address associate with the user |
| integration | string | Authentication integration module |
| statistics | string | |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |

| | | |
|-----------------|--------------|--|
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Delete

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Get

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |
| start | integer | (optional) (<i>default: 0</i>) Number of start data |
| limit | integer | (optional) (<i>default: 200</i>) Limit data count |

NOTES

For get method, if **{{value}}** not provided then Kiwire will return all available items subject to **start** and **limit**.

KIWIRE 2.0 – API2

MODULE: VOUCHER ACCOUNT

URL

[POST] `http://{{kiwire_ip}}/apiv2/voucher/{{method}}/{{value}}`

Where `{{method}}` can be;

- create - delete - get

Where `{{value}}` can be;

- voucher code

PARAMETERS REQUIRED

Create

| Parameter | Value | Description |
|-----------------|--------------|---|
| prefix | string | (optional) (<i>default: V_</i>) Prefix for every voucher generated |
| qty | integer | (optional) (<i>default: 10</i>) Quantity of voucher generated |
| plan | string | (optional) (<i>default: wx_temp</i>) The profile associated with voucher to be generated |
| nas | string | (optional) Apply zone restriction to the voucher to be generated, to be able to login from specific zone only |
| expiry | date | (optional) (<i>default: {{tomorrow date}}</i>) The validity of voucher to be generated |
| remark | string | (optional) (<i>default: {{tomorrow date}}</i>) Remark |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Delete

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Get

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |
| start | integer | (optional) Number of start data (<i>default: 0</i>) |
| limit | integer | (optional) Limit data count (<i>default: 200</i>) |

NOTES

For get method, if **{{value}}** not provided then Kiwire will return all available items subject to **start** and **limit**.

KIWIRE 2.0 – API2

MODULE: NAS

URL

[POST] `http://{{kiwire_ip}}/apiv2/nas/{{method}}/{{value}}`

Where `{{method}}` can be;

- create - delete - get

Where `{{value}}` can be;

- nas identity

PARAMETERS REQUIRED

Create

| Parameter | Value | Description |
|--------------------|--------------|---|
| nas-ip | string | IP Address of the NAS devices |
| nas-id | string | A unique name to identify the NAS equipment |
| type | string | NAS device type |
| ports | integer | The ' <i>change of authority</i> ' (COA) listening port of the nas device |
| secret | string | The Secret password that the NAS will use to communicate with Kiwire |
| server | string | Server |
| community | string | SNMP Community string |
| description | string | The description of the NAS equipment |
| location | string | Location of the NAS |
| username | string | The administrator username that is use to manage and monitor the NAS |
| password | string | The password for the administrator username used |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |

| | | |
|-----------------|--------|--|
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Delete

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Get

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |
| start | integer | (optional) Number of start data (<i>default: 0</i>) |
| limit | integer | (optional) Limit data count (<i>default: 50</i>) |

NOTES

For get method, if **{{value}}** not provided then Kiwire will return all available items subject to **start** and **limit**.

KIWIRE 2.0 – API2

MODULE: DEVICES

URL

[POST] `http://{{kiwire_ip}}/apiv2/device/{{method}}/{{value}}`

Where `{{method}}` can be;

- create - delete - get

Where `{{value}}` can be;

- voucher code

PARAMETERS REQUIRED

Create

| Parameter | Value | Description |
|--------------------|--------------|---|
| device-ip | string | IP Address of the equipment |
| device-id | string | A unique name to identify the equipment |
| type | string | Device type |
| mac | integer | The <i>'change of authority'</i> (COA) listening port of the nas device |
| community | string | SNMP Community string |
| description | string | The Description for the device/equipment |
| location | string | Location of the device/equipment |
| username | string | The username use to access the equipment remotely |
| password | string | The password used by the username to access the equipment |
| snmpv | integer | Version |
| mib | string | Management Information Base |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Delete

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Get

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |
| start | integer | (optional) Number of start data (<i>default: 0</i>) |
| limit | integer | (optional) Limit data count (<i>default: 50</i>) |

NOTES

For get method, if **{{value}}** not provided then Kiwire will return all available items subject to **start** and **limit**.

KIWIRE 2.0 – API2

MODULE: PROFILE

URL

[POST] `http://{{kiwire_ip}}/apiv2/profile/{{method}}/{{value}}`

Where {{method}} can be;

- create - delete - get

Where {{value}} can be;

- profile name

PARAMETERS REQUIRED

Create

| Parameter | Value | Description |
|---------------------|---------------|--|
| profile-name | string | The name of the profile/plan |
| price | float | The price of the plan |
| type | string | Type of plan |
| edx | string | Integration to charge to or link with |
| minute | integer | The allowed time (minutes) |
| vol_limit_en | 'on' or 'off' | Enable or disable volume transfer limit |
| vol_limit | integer | The limit of volume transfer (GB) |
| iddle | integer | Iddle timeout of user to disconnected by system |
| simultaneous | integer | Count of simultaneous user that use the same account associated with this plan |
| bwdown | integer | Maximum download speed allowed for user (Kbps) |
| bwup | integer | Maximum upload speed allowed for user (Kbps) |
| min_up | integer | Minimum download speed allowed for user (Kbps) |

| | | |
|-----------------|--------------|---|
| min_down | integer | Minimum upload speed allowed for user (Kbps) |
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Delete

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

Get

| Parameter | Value | Description |
|-----------------|--------------|---|
| api_key | string | MD5 hashed in this format: {real_api_key_from_kiwire}+{token}+{time} |
| time | date(YmdHis) | Current time Must be in this format: YmdHis |
| token | string | Token generated during the API login process |
| cloud_id | string | Cloud ID or Tenant name of the API key |

NOTES

For get method, if **{{value}}** not provided then Kiwire will return all available items subject to **start** and **limit**.

KIWIRE 2.0 – API2

SAMPLE API CODE

```
<?php
session_start();

// Configuration
$cloud_id = "default";
$username = "admin";
$password = "youradminpassword";
$sapi_key = "Nsf7a86787asvh&5asfkasdfNDJFNKsa=sS"; // API key get from Kiwire

$kiwire_host = "http://kiwireserveraddress.com";
$kiwire_port = 8081;

/* *****
** -- START --
** LOGIN & TOKEN PART
*/

// If token already exist, check if the token expired or not
if(isset($_SESSION['login_token']) && isset($_SESSION['login_last_time'])){

    $time_current = time();
    $time_login_expired = strtotime("+20 minutes", $_SESSION['login_last_time']);

    if($time_current < $time_login_expired) $do_not_get_token = true;

}
```



```

// If token not exist or token expired, get new token
if(empty($do_not_get_token)){

    // setup api_login method to get the token
    $login_api_data = Array(
        "url"    => $kiwire_host . "/admin/agent/api_login.php",
        "port"   => $kiwire_port,
        "params" => Array(
            "code" => MD5( $username . "|" . $password ),
            "cloud_id" => $cloud_id,
        ),
    );

    // execute api_login method to get the token
    $result_json = sendhttpreq($login_api_data['url'], $login_api_data['params'], "POST", $login_api_data['port']);
    $result_array = json_decode($result_json, true);

    $_SESSION['login_token'] = $result_array['token']; //the token will expired in 20 minutes and must renew the token each 20 minutes
    $_SESSION['login_last_time']= time();

}

/*
** -- END --
** LOGIN & TOKEN PART
** ***** */

// this $current_datetime variable
// must be included in the hashed API Key ( MD5($real_api_key . $login_token . $current_datetime) )
// also must be included in params as "time"
// - refer the code below
$current_datetime = date("YmdHis");

```

```
// set up login history method

// you can refer to documentation for another method

$api_call_data = Array(

    "url"    => $kiwire_host . "/apiv2/login/history",

    "port"   => $kiwire_port,

    "params" => Array(

        "start_time" => "2017-06-08 00:00:00",

        "end_time"   => "2017-06-08 23:59:59",

        "token"      => $_SESSION['login_token'],

        "api_key"    => MD5( $api_key . $_SESSION['login_token'] . $current_datetime ),

        "time"       => $current_datetime,

        "cloud_id"   => $cloud_id,

    ),

);

// execute login history method

$result_json = sendhttpreq($api_call_data['url'], $api_call_data['params'], "POST", $api_call_data['port']);

$result_array = json_decode($result_json, true);

// if login token expired, refresh to get new token

if(substr($result_array['error'], 0, 13) == "login expired"){

    session_destroy();

    echo "<script>window.location.reload();</script>";

}

// display the result

var_dump($result_array);
```

```
// Function for sending HTTP Request
function sendhttpreq($url, $data, $method="GET", $port=80){
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_PORT, $port);
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, $method);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

    $result = curl_exec($ch);
    curl_close($ch);

    return $result;
}
?>
```

KIWIRE 2.0 – API2

LIST OF ERROR MESSAGE

| Error Message | Cause | Solution |
|---|---|--|
| permission denied | Missing token or api_key or cloud_id | Please include fields token, api_key and cloud_id in all your request. |
| login expired. please complete the api login again | Token has expired | Must login back via API to get new token |
| please provide full url | URL is not valid | Please refer documentation for valid URL |
| invalid api key | API key or hashed API key is incorrect | Make sure the API key, token, and time is correct |
| api integration is currently disabled | API integration is OFF | Make sure the API integration is ON in Kiwire UI platform |
| module not existed | URL or method is not valid | Please refer documentation for valid URL or method |
| module disabled | Selected module is OFF | Make sure the module is ON in Kiwire UI platform |
| no tenant / cloud id provided | No Cloud ID provided | Please include cloud_id parameter in every API HTTP request |
| cloud id not found | Cloud ID provided is not valid | Make sure the Cloud ID provided is valid |
| cloud id not available | Cloud ID provided is not valid | Make sure the Cloud ID provided is valid |
| not valid user | Username or Password or Cloud ID is not matched | Please use correct username, password and cloud ID |