

ISP 管理

ISP 相關 API

Resource

URI	Method	Params	Response
/owl/isps List of ISPs <pre>curl http://<host>/owl/isps</pre>	GET	* name for ISPs whose isp_name fields match the query	200 , <array of isp>
/owl/isp/{isp_id} Get data of a ISP <pre>curl http://<host>/owl/isp/{isp_id}</pre>	GET	* {isp_id} - The id of ISP to be retrieved	200 , <isp>
/owl/isp Add a new ISP <pre>curl -X POST -H "Content-Type: application/json" -d '{"name":"isp-1","acronym":"ISP1"}' http://<host>/owl/isp</pre>	POST	* <isp modify> - The body of ISP	201 , <isp>
/owl/isp/{isp_id} Modify an existing host <pre>curl -X PUT -H "Content-Type: application/json" -d '{"name":"isp-1","acronym":"ISP1"}' http://<host>/owl/isp/{isp_id}</pre>	PUT	<ul style="list-style-type: none">{isp_id} - The id of ISP to be modified<isp modify> - The body of ISP	200 , <isp>

ISP 管理
省份 管理
城市 管理
Name Tag
Group Tag

Data

Name	Content	Detail		
		Name	Constraint	Comment
<isp>	{ " id " : 32, " name " : " 方正宽 带 ", " acronym " : " FBN " }	name	Mandatory	The name of ISP
		acronym	Mandatory	The acronym of ISP
		id	[auto generated]	The id of ISP

Name	Content	Detail									
<code><isp modify></code>	<pre>{ "name": "方正宽 带", "acronym": "FBN" }</pre>	<table> <tr> <th>Name</th><th>Constraint</th><th>Comment</th></tr> <tr> <td><code>name</code></td><td>Mandatory</td><td>The name of ISP</td></tr> <tr> <td><code>acronym</code></td><td>Mandatory</td><td>The acronym of ISP</td></tr> </table>	Name	Constraint	Comment	<code>name</code>	Mandatory	The name of ISP	<code>acronym</code>	Mandatory	The acronym of ISP
Name	Constraint	Comment									
<code>name</code>	Mandatory	The name of ISP									
<code>acronym</code>	Mandatory	The acronym of ISP									
<code><array of isp></code>	<pre>[{ /* See <isp> */ }, /* More hosts... */]</pre>	see <code><isp></code>									

Errors

Status	Content	comment
409	<pre>{ "error_code": 1, "error_message": "ISP acronym is existing" }</pre>	<code>error_code</code> : <ul style="list-style-type: none"> 1 - ISP acronym is existing(duplicated)

省份 管理

省份相關 API

Resource

URI	Method	Params	Response
<code>/owl/provinces</code> List of provinces <pre>curl http://<host> /owl/provinces</pre>	GET	* <code>name</code> for provinces whose pv_name fields match the query	200 , <code><array of province></code>
<code>/owl/province/{province_id}</code> Get data of a province <pre>curl http://<host> /owl/province/{province_id}</pre>	GET	* <code>{province_id}</code> - The id of province to be retrieved	200 , <code><province></code>
<code>/owl/province</code> Add a new province <pre>curl -X POST -H "Content-Type: application/json" -d '{"name":"province-1"}' \ http://<host> /owl/province</pre>	POST	* <code><province modify></code> - The body of province	201 , <code><province></code>
<code>/owl/province/{province_id}</code> Modify an existing host <pre>curl -X PUT -H "Content-</pre>	PUT	<ul style="list-style-type: none"> <code>{province_id}</code> - The id of province to be modified <code><province modify></code> - The 	200 , <code><province></code>

URI	Method	Params	Response
Type: application/json" -d '{"name":"province-1"}' \ http://<host> /owl/province/{province_id}		body of province	

Data

Name	Content	Detail		
<province>	<pre>{ "id" : 32, "name": "廣東省" }</pre>	Name	Constraint	Comment
		id	[auto generated]	The id of province
		name	Mandatory	The name of province
<province modify>	<pre>{ "name": "廣東省" }</pre>	Name	Constraint	Comment
		name	Mandatory	The name of province
<array of province>	<pre>[{ /* See <province> */ }, /* More hosts... */]</pre>	see <province>		

Errors

Status	Content	comment
409	<pre>{ "error_code": 1, "error_message": "province name is existing" }</pre>	error_code: <ul style="list-style-type: none"> 1 - province name is existing(duplicated)

城市 管理

城市相關 API

Resource

URI	Method	Params	Response
<owl/cities> List of cities <pre>curl http://<host> /owl/cities</pre>	GET	* <name> for cities whose ct_name fields match the query	200 , <array of cities1>
<owl/province/{province_id}> <cities> List of cities for a province <pre>curl http://<host> /owl/province /{province_id}/cities</pre>	GET	<ul style="list-style-type: none"> <province_id> - The id of province <name> for cities whose ct_name fields match the query 	200 , <array of cities2>

URI	Method	Params	Response
/owl/city/{city_id} Get data of a city <pre>curl http://<host>/owl/city/{city_id}</pre>	GET	* {city_id} - The id of city to be retrieved	200 , <city>
/owl/city Add a new city <pre>curl -X POST -H "Content-Type: application/json" \ -d '{"name":"city-1", "province_id": 7, "post_code": "12938477"}' http://<host>/owl/city</pre>	POST	* <city modify> - The body of city	201 , <city>
/owl/city/{city_id} Modify an existing host <pre>curl -X PUT -H "Content-Type: application/json" \ -d '{"name":"city-1", "province_id": 7, "post_code": "12938477"}' http://<host>/owl/city/{city_id}</pre>	PUT	<ul style="list-style-type: none"> * {city_id} - The id of city to be retrieved * <city modify> - The body of city 	200 , <city>

Data

Name	Content	Detail		
<div><city></div>	<div><pre>{ "id" : 32, "name": "中山市", "post_code": "23423", "province": { "id" : 3, "name" : "浙江省" } }</pre></div>			
		Name	Constraint	Comment
		id	[auto generated]	The id of city
		name	Mandatory	The name of city
		post_code	Mandatory	The post code of city
province	Mandatory	The province of city		
<div><city modify></div>	<div><pre>{ "name": "中山市", "post_code": "23423", "province_id": 32 }</pre></div>			
		Name	Constraint	Comment
		name	Mandatory	The name of city
		post_code	Mandatory	The post code of city
		province_id	Mandatory	The id of province for this city

Name	Content	Detail
<code><array of city1></code> , has <code>province</code> property	<pre>[{ /* See <city> */ }, /* More cities... */]</pre>	See <code><city></code>
<code><array of city2></code> , without <code>province</code> property	<pre>[{ "id" : 32, "name": "中山市", "post_code": "23423", }, /* More cities... */]</pre>	See <code><city></code>

Errors

Status	Content	comment
409	<pre>{ "error_code": 1, "error_message": "post code of city is existing" }</pre>	<p><code>error_code</code>:</p> <ul style="list-style-type: none"> 1 - post code of city is existing(duplicated) 2 - same name of city for a province

Name Tag

設備機房，用於查詢過濾用。

Resources

URI	Method	Params	Response
/owl/nametags List of Name Tags <pre>curl http://<host> /owl/nametags</pre>	GET	<ul style="list-style-type: none"> <code>value</code> for Name Tags whose nt_value fields match the query <code>page-size</code> - 50 <code>page-pos</code> - 1 <code>total-count</code> - N SORTING: <code>value</code> 	200 , <code><array of name tags></code>
/owl/nametag/{nt_id} Get data of a Name Tag <pre>curl http://<host> /owl/nametag/{nt_id}</pre>	GET	<ul style="list-style-type: none"> * <code>{nt_id}</code> - The id of name tag to be retrieved 	200 , <code><name tag></code>
/owl/nametag TODO Add name tag	POST	<ul style="list-style-type: none"> * <code><modify body of name tag></code> - The body of name tag 	200 , <code><body of name tag></code>

URI	Method	Params	Response
<pre>curl -d '{"value":"电信_北京","ids_of_agents":[],"ids_of_targets":[]}' http://<host>/owl/nametag</pre>			
<p>/owl/nametag/{nametag_id}</p> <p>TODO Sets data of name tag</p> <pre>curl -X PUT -d '{"value":"电信_北京","ids_of_agents":[],"ids_of_targets":[]}' http://<host>/owl/nametag/7612</pre>	PUT	<ul style="list-style-type: none"> {nametag_id} - The id of name tag <modify body of name tag> - The body of name tag 	200 , <body of name tag>

Data

Name	Content	Detail															
<name tag>	<pre>{ "id" : 32, "value" : "电信_北京" }</pre>	<table> <tr> <th>Name</th><th>Constraint</th><th>Comment</th></tr> <tr> <td>id</td><td>Mandatory</td><td>The id of name tag</td></tr> <tr> <td>value</td><td>Mandatory [unique]</td><td>The value of name tag</td></tr> </table>	Name	Constraint	Comment	id	Mandatory	The id of name tag	value	Mandatory [unique]	The value of name tag						
Name	Constraint	Comment															
id	Mandatory	The id of name tag															
value	Mandatory [unique]	The value of name tag															
<array of name tags>	<pre>[/* see <name tag> */]</pre>	see <name tag>															
<body of name tag> TODO	<pre>{ "id" : 32, "value" : "电信_北京", "agents" : [{ "id" : 332, "name" : "pdc-jk-123-061-105-009" }, /* More agents... */], "targets" : [{ "id": 332, "name": "pdc-jk-123-061-105-009", "host": "123.61.105.9" }, /* More targets... */] }</pre>	<table> <tr> <th>Name</th><th>Constraint</th><th>Comment</th></tr> <tr> <td>id</td><td>Mandatory</td><td>The id of name tag</td></tr> <tr> <td>value</td><td>Mandatory [unique]</td><td>The value of name tag</td></tr> <tr> <td>agents</td><td>[empty]</td><td>The data of agents in this name tag</td></tr> <tr> <td>targets</td><td>[empty]</td><td>The data of targets in this name tag</td></tr> </table>	Name	Constraint	Comment	id	Mandatory	The id of name tag	value	Mandatory [unique]	The value of name tag	agents	[empty]	The data of agents in this name tag	targets	[empty]	The data of targets in this name tag
Name	Constraint	Comment															
id	Mandatory	The id of name tag															
value	Mandatory [unique]	The value of name tag															
agents	[empty]	The data of agents in this name tag															
targets	[empty]	The data of targets in this name tag															

Name	Content	Detail		
<code><modify body of name tag></code> TODO	<pre>{ "value" : "电信_北京", "ids_of_agents" : [13, 451, 98], "ids_of_targets" : [16, 778, 91] }</pre>	Name	Constraint	Comment
		value	Mandatory [unique]	The value of name tag
		ids_of_agents	[empty]	The data of agents in this name tag
		ids_of_targets	[empty]	The data of targets in this name tag

Errors

Status	Content	comment
409	<pre>{ "error_code": 1, "error_message": "Value of name tag is existing" }</pre>	error_code : <ul style="list-style-type: none"> 1 - value of name tag is existing

Group Tag

設備群組，用於查詢過濾用。每台機器都有 ISP、地點、機房等同通屬性，不同團隊要看的資料需用 Group Tag 方式區隔。

Resources

URI	Method	Params	Response
/owl/grouptags List of Group Tags <pre>curl http://<host>/owl/grouptags</pre>	GET	<ul style="list-style-type: none"> name for Group Tags whose gt_name fields match the query page-size - 50 page-pos - 1 total-count - N SORTING: name 	200 , <array of group tags>
/owl/grouptag/{gt_id} Get data of a Group Tag <pre>curl http://<host>/owl/grouptag/{gt_id}</pre>	GET	* {gt_id} - The id of group tag to be retrieved	200 , <group tag>
/owl/grouptag TODO Add group tag <pre>curl -d '{"name":"boss","ids_of_agents":[],"ids_of_targets":[]}' http://<host>/owl/grouptag</pre>	POST	* <code><modify body of group tag></code> - The body of group tag	200 , <body of group tag>

URI	Method	Params	Response
/owl/grouptag/{grouptag_id} TODO Sets data of group tag <pre>curl -X PUT -d '{"name":"boss","ids_of_agents": [],"ids_of_targets":[]}' http://<host>/owl/grouptag/7612</pre>	PUT	<ul style="list-style-type: none"> {grouptag_id} <ul style="list-style-type: none"> - The id of group tag <modify body of group tag> <ul style="list-style-type: none"> - The body of group tag 	200 , <body of group tag>

Data

Name	Content	Detail		
<group tag>	<pre>{ "id" : 32, "name" : "BOSS" }</pre>	Name	Constraint	Comment
		id	Mandatory	The id of group tag
		name	Mandatory [unique]	The name of group tag
<array of group tags>	[/* see <group tag> */]	see <group tag>		
<body of group tag> TODO	<pre>{ "id" : 32, "name" : "BOSS", "comment" : "This is group tag", "agents" : [{ "id" : 332, "name" : "pdc-jk-123-061-105-009" }, /* More agents... */], "targets" : [{ "id": 332, "name": "pdc-jk-123-061-105-009", "host": "123.61.105.9" }, /* More targets... */] }</pre>	Name	Constraint	Comment
		id	Mandatory	The id of group tag
		name	Mandatory [unique]	The name of group tag
		comment		The comment of group tag
		agents	[empty]	The data of agents in this group tag
		targets	[empty]	The data of targets in this group tag
<modify body of group tag> TODO	<pre>{ "name" : "BOSS", "comment" : "This is group tag", "ids_of_agents" : [13, 451, 98], "ids_of_targets" : [16, 778, 91] }</pre>	Name	Constraint	Comment
		name	Mandatory [unique]	The name of group tag
		comment		The comment of group tag

Name	Content	Detail		
		Name	Constraint	Comment
		ids_of_agents	[empty]	The data of agents in this group tag
		ids_of_targets	[empty]	The data of targets in this group tag

Errors

Status	Content	comment
409	<pre>{ "error_code": 1, "error_message": "Name of group tag is existing" }</pre>	<p>error_code :</p> <ul style="list-style-type: none"> 1 - name of group tag is existing

Restful API 設計慣例

若要增加 header, 請記得在 CORS 中增加對應的 `Access-Control-Allow-Headers`

See [Cors Access-Control-Allow-Headers wildcard being ignored?](#)

API 版本

使用 `URI` 代表版本:

1. 沒有版號預設為 `v1` 版, 例: `http://<host>/api/v1/<resource...>`

其它 versioning 方式

- 使用 domain name - `<version>.service`, 適合用在另一版用另一程式語言實作
- Header - 讓 URI 維持不動
- Query Parameter - 讓 URI 維持不動

使用何種機制來指定 API 的 Version 是 [哲學問題](#)

頁碼分頁

當 API 要提供分頁時, 一律以下列規格設計參數

在文件使用 **paging-number**, 代表此 API 有分頁功能

在 **HTTP Header** 中, 自定義下列 Header:

Request

- `page-size` - 每頁筆數, API 一定要提供預設值(值過大會以可允最大值覆寫)
- `page-pos` - 第 N 頁, 從 1(預設值) 開始
- `order-by` - `<prop_1>#<dir>:<prop_2>#<dir>` - 以 `:` 分隔的多重欄位排序語法
 - `<prop_1>` - 屬性名稱
 - `<dir>` - 方向
 - 範例: `name#asc:connection_id:status#desc` - 先以 **name** 排序(遞增), 再以 **connection_id**(預設方向), 最後以 **status** 排序(遞減)

Response

- `page-size` - 每頁筆數，Server 回應的安全值
- `page-pos` - Server 回應的安全值
- `total-count` - 總筆數，API 可不提供此 header
- `page-more` - `true`, `false` 當總筆數會影響效能時，此 header 代表是否有下一頁

使用 Header 的好處是，輸出多筆資料的 API 不需使用 JSON Object 的格式

指標分頁

在文件使用 **paging-pointer**，代表此 API 有分頁功能

指標應包含排序資訊等分頁額外資料

在 **HTTP Header** 中，自定義下列 Header:

Request

- `page-ptr: <location>` - 可以是 URL, HASH code 或 URI，代表開始的指標

Response

- `previous-page: <location>` - 可以是 URL, HASH code 或 URI，代表前一頁的指標
- `next-page: <location>` - 可以是 URL, HASH code 或 URI，代表下一頁的指標

複雜分頁

由 API 的 JSON 內容自行定義

Parameters

關於參數，或資料欄位的屬性

- **Mandatory** - Client 一定要提供的資料
- `[auto-generated]` - 系統自動產生的資料，例如 internal key，**不能被新增或修改**
- `[new]` - 只能被新增，不能被修改
- `[null]` - Server 可能傳回 null 值
- `[missed]` - 這個屬性可能完全不顯示出來
- `[empty]` - JSON 的 array 可能是空陣列 `"hosts" : []`
- `[unix time]` - 值為 **UNIX Time**, 例: `1471995721` (2016-08-23T23:42:01Z)

全域錯誤

當 API 有下列錯誤時，一律以下列規格產生結果

Response **400**

(`application/json`) 資料驗證錯誤，例如密碼太短或必需要填入非空白字串通常是 Client Side 就能檢查的錯誤

```
{
  "http_status": 400,
  "error_code" : -1,
  "error_content" : [
    { /* 屬性的名稱與錯誤原因 */ }
  ]
}
```

Response **409**

(`application/json`) 唯一性錯誤，必需由 Server Side 檢查

```
{
  "http_status": 409,
  "error_code" : -1,
  "error_content" : [
    { /* 屬性的名稱與錯誤原因 */ }
  ]
}
```

Response 500

(application/json) 非預期的錯誤，例資料庫無法連到或 go lang panic

```
{
  "http_status": 500,
  "error_code" : -1,
  "error_message": "dial tcp 192.168.20.50:3306: connectex: No connection
...", // 錯誤訊息
  "error_stack": [
    "github.com/Proj/a.go:458",
    "github.com/Proj/a.go:332"
  ]
}
```

Response 404

(application/json) 資源不存在

```
{
  "http_status": 404,
  "error_code" : -1,
  "uri": "/no-such-resource"
}
```

Response 405

(application/json) 不允許使用該方法(method)

```
{
  "http_status": 405,
  "error_code" : -1,
  "method": "POST",
  "uri": "/no-such-resource"
}
```

Response 401

(application/json) 沒有登入的錯誤，可能用 404 來提高安全度

```
{
  "http_status": 401,
  "error_code" : -1
}
```

Response 403

(application/json) 有登入(系統知道你是誰)，但沒有權限存取資源可能用 404 來提高安全度

```
{
  "http_status": 403,
  "error_code" : -1
}
```

錯誤分層

第一層: HTTP Status

第二層: Response Body

例：下列例子中，error_code 為 1 有兩種，分別以 400 與 401 區分

Status	JSON Body	Comment
400	{ "error_code" : 1 }	400 AND 1
401	{ "error_code" : 1 }	401 AND 1
401	{ "error_code" : 2 }	401 AND 2

Last modified on 2016-12-05T17:52:35+08:00