

IGEL Management Interface (IMI)

Exported on 10/26/2021



Table of Contents

1	IMI Manual	12
1.1	Licensing.....	12
1.1.1	UMS 5.09.100 or Older.....	12
1.1.2	UMS 6.01.100 or Newer	13
1.2	REST Basics	13
1.2.1	Ressources.....	13
	URLs Represent Resources	13
	Further Examples of Resources	13
1.2.2	HTTP Methods	14
1.3	Prerequisites	14
1.3.1	IGEL Universal Management Suite (UMS)	14
1.3.2	Networking.....	14
1.4	First Steps	15
1.4.1	Client Applications and Libraries	15
	Clients	15
	Programming libraries	17
1.4.2	Getting the Server Status.....	17
	Response Headers.....	17
	JSON Response Body.....	17
1.4.3	Authentication	18
	Logging In	18
	Staying Logged in.....	18
1.4.4	Listing all Thin Clients.....	18
	Response	19
1.4.5	Getting Information on a Thin Client	20
	Response	21
1.4.6	Getting All Details about a Thin Client	21
	Response	22
1.4.7	Getting Thin Client Status.....	24
	Response	24
1.5	Creating, Updating and Deleting Resources.....	25
1.5.1	Listing all Thin Client Directories	25
	Response	25



1.5.2 Creating a Thin Client Directory	27
Response	28
1.5.3 Updating a Thin Client Directory	28
Response	28
1.5.4 Deleting a Thin Client Directory	29
Response	29
1.6 Further Operations	30
1.6.1 Moving a Thin Client Directory	30
Response	30
1.6.2 Sending a Command to Thin Clients.....	31
Request.....	31
1.6.3 Debugging Requests	32
Error Codes.....	32
HTTP-Proxy.....	32
2 Powershell.....	34
2.1 Introduction	34
2.1.1 Installation	34
Prerequisites	34
Installing.....	34
2.1.2 Loading the IMI Cmdlets Snap-in	35
2.1.3 Authentication	35
Parameters	35
Example Command Line.....	36
2.2 Command Reference	36
2.2.1 Thin Client	36
Get-IgelTCInformation	36
Start-IgelTCs.....	38
Stop-IgelTC.....	38
Restart-IgelTCs	39
Move-IgelTCs	39
Update-IgelTCSettings.....	40
2.2.2 Profile	41
Get-IgelTCProfile	41
Rename-IgelTCProfile.....	41
Get-IgelProfileAssignments	42



Set-IgelProfileAssignment	43
Remove-IgelProfileAssignment	44
Remove-IgelTCProfile	45
2.2.3 Master Profile	45
Get-IgelTCMasterprofile.....	45
Rename-IgelTCMasterprofile.....	46
Get-IgelMasterprofileAssignments.....	47
Set-IgelProfileAssignment	47
Remove-IgelProfileAssignment	48
Remove-IgelMasterprofile	49
2.2.4 Thin Client Directory	50
Get-IgelTCDirectories.....	50
Rename-IgelTCDirectory	51
Set-IgelTCDirectory	52
Move-IgelTCDirectory	52
Set-IgelProfileAssignment	53
Remove-IgelTCDirectory.....	54
2.2.5 Profile Directory	54
Get-IgelProfileDirectory	55
Rename-IgelTCProfileDirectory	55
Move-IgelTCProfileDirectory	56
Set-IgelProfileDirectory	56
Remove-IgelProfileDirectory	57
2.2.6 Master Profile Directory	58
Get-IgelMasterprofileDirectory.....	58
Rename-IgelTCMasterprofileDirectory	59
Move-IgelMasterprofileDirectory	59
Set-IgelMasterprofileDirectory	60
Remove-IgelMasterprofileDirectory	60
2.2.7 Firmware.....	61
Get-IgelFirmware	61
2.2.8 Server Status	62
Get-IgelRMServerStatus.....	62
3 IMI API V3 Reference.....	63
3.1 What's New in IMI V3?	63



3.2 Prerequisites	63
3.3 Client Applications and Libraries	63
3.3.1 Clients	63
3.3.2 Programming	65
3.4 Authentication	65
3.4.1 POST /v3/login	65
Summary	65
Resource URL	65
Example Request	65
Response Body	65
Example Response	66
3.4.2 POST /v3/logout	66
Summary	66
Resource URL	66
Example Request	66
Example Response	66
3.5 Basic Data Types	67
3.5.1 ApiObject	67
Summary	67
Properties	67
Example	67
3.5.2 AssetInfoResource	68
Summary	68
Properties	68
Example	68
3.5.3 AssetDetailResource	69
Summary	69
Properties	69
Example	69
3.5.4 AssetCategory	70
Summary	70
Properties	70
Example	70
3.5.5 AssetHistoryResource	70
Summary	70



Properties	70
Example	71
3.5.6 TCResourceV3	71
Summary	71
Properties	71
With facets=details.....	72
With facets=online	74
With facets=shadow	74
3.5.7 Assignment.....	74
Summary	74
Properties	75
Example	75
3.5.8 DirectoryResource	76
Properties	76
With facets=children	76
3.6 Resources	77
3.6.1 Asset Information.....	77
GET /v3/assetinfo	77
GET /v3/assetinfo/assets/{assetId}	81
GET /v3/assetinfo/thinclients/{tclId}	83
GET /v3/assethistory	85
GET /v3/assethistory/assets/{assetId}	88
GET /v3/assethistory/thinclients/{unitId}.....	89
3.6.2 Thin Client	91
GET /v3/thinclients	91
GET /v3/thinclients?facets=details	93
GET /v3/thinclients?facets=online	96
GET /v3/thinclients?facets=shadow.....	98
GET /v3/thinclients/{tclId}	100
GET /v3/thinclients/{tclId}?facets=details	101
GET /v3/thinclients/{tclId}?facets=online	103
GET /v3/thinclients/{tclId}?facets=shadow	104
GET /v3/thinclients/{tclId}/assignments/profiles.....	106
PUT /v3/thinclients/{tclId}/assignments/profiles	109
PUT /v3/thinclients	110



PUT /v3/thinclients/{tcId}	111
DELETE /v3/thinclients/{tcId}	113
DELETE /v3/thinclients/{tcId}/deletetcoffline	114
POST /v3/thinclients/?command={reboot shutdown wakeup}	115
POST /v3/thinclients?command=settings2tc	116
POST /v3/thinclients/?command=tcreset2facdefs	117
3.6.3 Profile	118
GET /v3/profiles	118
GET /v3/profiles/{profileid}	120
PUT /v3/profiles/{profileid}	121
DELETE /v3/profiles/{profileid}	122
GET /v3/profiles/{profileid}/assignments/thinclients	123
PUT /v3/profiles/{profileid}/assignments/thinclients/	125
DELETE /v3/profiles/{profileid}/assignments/thinclients/{id}	126
GET /v3/profiles/{profileid}/assignments/tcdirectories	127
PUT /v3/profiles/{profileid}/assignments/tcdirectories	129
DELETE /v3/profiles/{profileid}/assignments/tcdirectories/{id}	130
3.6.4 Master Profile	131
GET /v3/masterprofiles	131
GET /v3/masterprofiles/{profileid}	133
PUT /v3/masterprofile/{profileid}	134
DELETE /v3/masterprofiles/{profileid}	135
GET /v3/masterprofiles/{profileid}/assignments/thinclients	136
PUT /v3/masterprofiles/{profileid}/assignments/thinclients/	138
DELETE /v3/masterprofiles/{profileid}/assignments/thinclients/{id}	140
GET /v3/masterprofiles/{profileid}/assignments/tcdirectories	140
PUT /v3/masterprofiles/{profileid}/assignments/tcdirectories/	142
DELETE /v3/masterprofiles/{profileid}/assignments/tcdirectories/{id}	143
3.6.5 Thin Client Directory	144
GET /v3/directories/tcdirectories	144
GET /v3/directories/tcdirectories?facets=children	146
GET /v3/directories/tcdirectories/{id}	149
GET /v3/directories/tcdirectories/{id}?facets=children	150
PUT /v3/directories/tcdirectories/	153
PUT /v3/directories/tcdirectories/{id}	154
PUT /v3/directories/tcdirectories/{id}?operation=move	155



DELETE /v3/directories/tcdirectories/{id}	156
GET /v3/directories/tcdirectories/{id}/assignments/profiles	157
PUT /v3/directories/tcdirectories/{id}/assignments/profiles.....	159
3.6.6 Profile Directory	160
GET /v3/directories/profiledirectories	161
GET /v3/directories/profiledirectories?facets=children	163
GET /v3/directories/profiledirectories{id}	167
GET /v3/directories/profiledirectories{id}?facets=children.....	168
PUT /v3/directories/profiledirectories/	170
PUT /v3/directories/profiledirectories/{id}.....	171
PUT /v3/directories/profiledirectories/{id}?operation=move	172
DELETE /v3/directories/profiledirectories/{id}.....	173
3.6.7 Master Profile Directory	174
GET /v3/directories/masterprofiledirectories	174
GET /v3/directories/masterprofiledirectories?facets=children	177
GET /v3/directories/masterprofiledirectories/{id}	179
GET /v3/directories/masterprofiledirectories{id}?facets=children	180
PUT /v3/directories/masterprofiledirectories/	182
PUT /v3/directories/masterprofiledirectories/{id}.....	183
PUT /v3/directories/masterprofiledirectories/{id}?operation=move	183
DELETE /v3/directories/masterprofiledirectories{id}	185
3.6.8 Firmware Information.....	185
GET /v3/firmwares	185
GET /v3/firmwares/{fwId}	187
3.6.9 Server Information.....	188
GET /v3/serverstatus.....	188
3.7 Error Codes	189
3.7.1 HTTP Status Codes.....	189
Client-Side Errors	189
Server-Side Errors	189
3.7.2 API Error Messages.....	189
4 IMI How-Tos.....	193
4.1 Using Secure VNC via IGEL Management Interface (IMI)	193
4.1.1 Prerequisites	193
4.1.2 Thin Client Configuration	193



4.1.3 Secure VNC with IMv2	193
----------------------------------	-----

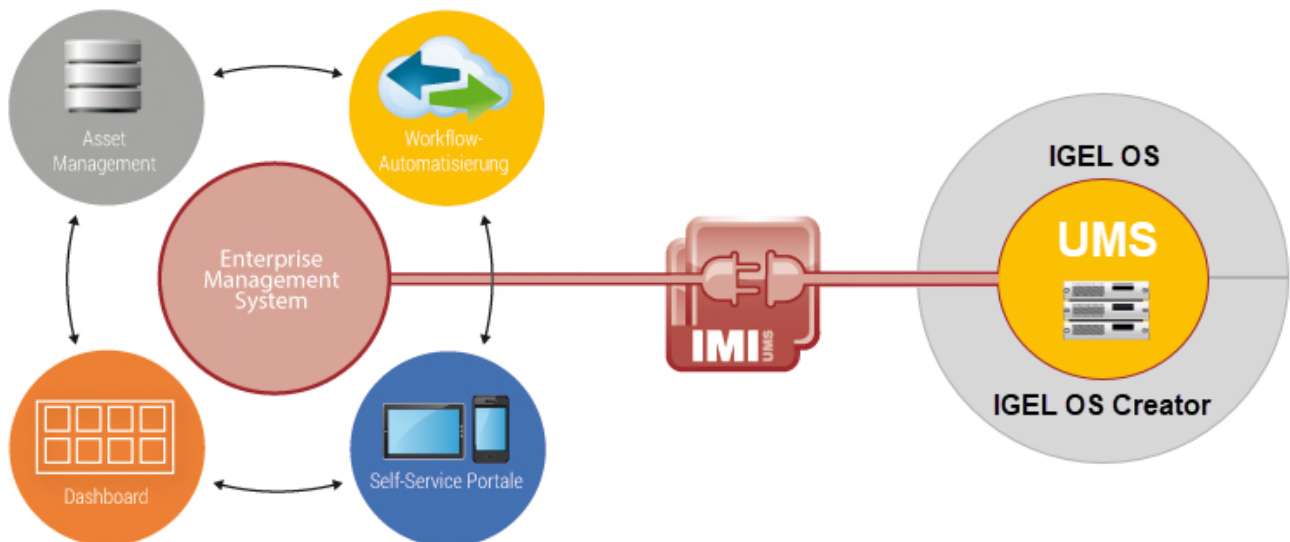


- [IMI Manual](#)(see page 12)
- [Powershell](#)(see page 34)
- [IMI API V3 Reference](#)(see page 63)
- [IMI How-Tos](#)(see page 193)



1 IMI Manual

IGEL Management Interface (IMI) enables you to connect UMS to systems management tools. It is a programming interface that can create and delete endpoint devices, move them between directories, reboot them and much more. Its implementation as a REST API makes IMI agnostic of hardware platforms, operating systems and programming languages, thus ensuring maximum interoperability.



This document serves as an introduction to using IGEL Management Interface (IMI).

Detailed information about the requests can be found in the [IMI API V3 Reference](#)(see page 63).

- [Licensing](#)(see page 12)
- [REST Basics](#)(see page 13)
- [Prerequisites](#)(see page 14)
- [First Steps](#)(see page 15)
- [Creating, Updating and Deleting Resources](#)(see page 25)
- [Further Operations](#)(see page 30)

1.1 Licensing

The licensing for IGEL Management Interface (IMI) depends on the UMS version you are using.

When Asset Inventory Tracker (AIT) has been licensed, you can use the resources `assetinfo` and `assethistory`.

1.1.1 UMS 5.09.100 or Older

For basic functionality, see [Licensing IMI](#)¹.

¹ <https://kb.igel.com/display/licensesmorelegacy/Licensing+IMI>



For Asset Inventory Tracker (AIT), see [Licensing AIT](#)².

1.1.2 UMS 6.01.100 or Newer

IMI is part of the IGEL Workspace Edition (WE); no additional license is necessary. For further information, see [Workspace Edition](#)³.

For Asset Inventory Tracker (AIT), licenses from the Enterprise Management Pack (EMP) are required. For further information, see [Enterprise Management Pack](#)⁴.

1.2 REST Basics

IGEL Management Interface uses *REST*, which stands for Representational State Transfer.

REST is an architectural style for client-server applications, mainly implemented in the HTTP(S) protocol. Therefore it can be used with all technologies that can send and receive HTTP requests.

REST establishes a typical pattern that helps programmers to understand the structure of individual APIs. Its most important concepts are [Resources](#)(see page 13) and [HTTP Methods](#)(see page 14).

1.2.1 Ressources

URLs Represent Resources

A *REST* API makes resources available at specific URLs. You can find a list of all thin clients in the UMS REST API at:

- `https://[server]:8443/umsapi/v3/thinclients`

In a shorter notation, which assumes the base URL to be known:

- `/v3/thinclients`

In order to address an individual instance of a resource - a thin client, for example - you specify its ID in the URL:

- `/v3/thinclients/8`

Further Examples of Resources

URL	Resource
<code>/v3/directories/tcdirectories</code>	A list of all thin client directories
<code>/v3/directories/tcdirectories/123</code>	The thin client directory with the ID 123

² <https://kb.igel.com/display/licensesmorelegacy/Licensing+AIT>

³ <https://kb.igel.com/display/licensesmoreigelos11/Workspace+Edition>

⁴ <https://kb.igel.com/display/licensesmoreigelos11/Enterprise+Management+Pack>



/v3/firmwares	A list of all firmwares in the UMS instance
/v3/firmwares/7	The firmware with the ID 7

Find a list of all available resources in the [IMI API Reference](#)(see page 63).

1.2.2 HTTP Methods

You call HTTP methods for resources in order to manipulate them. The REST architectural style has a conventional meaning for each of the HTTP methods, which are also called verbs. The methods are listed below:

HTTP Method	
GET	Read information from a resource.
PUT	Create a new resource or update an existing resource.
POST	(Create a new resource *), send a command.
DELETE	Delete a resource.

* There is a subtle semantic difference between PUT and POST, which is sometimes a matter of dispute. *IMI* API favors

- PUT for create and update actions and uses
- POST for logins and for sending commands to resources.

1.3 Prerequisites

1.3.1 IGEL Universal Management Suite (UMS)

UMS 5.07.100 or higher

1.3.2 Networking

In order to use *IGEL Management Interface* you need to be able to reach the API host via the network and connect to its API port, TCP 8443 by default.

The base URL is

- `https://[server]:8443/umsapi/`

The resources for IMI version 3 are available at:

- `https://[server]:8443/umsapi/v3/`



IMI uses HTTPS to ensure the integrity and confidentiality of network traffic. It is good practice to use a valid server certificate with a verifiable signature. For more information, see [Using Your Own Certificates for Communication over the Web Port \(Default: 8443\)](#)⁵⁶.

1.4 First Steps

- [Client Applications and Libraries](#)(see page 15)
- [Getting the Server Status](#)(see page 17)
- [Authentication](#)(see page 18)
- [Listing all Thin Clients](#)(see page 18)
- [Getting Information on a Thin Client](#)(see page 20)
- [Getting All Details about a Thin Client](#)(see page 21)
- [Getting Thin Client Status](#)(see page 24)

1.4.1 Client Applications and Libraries

Clients

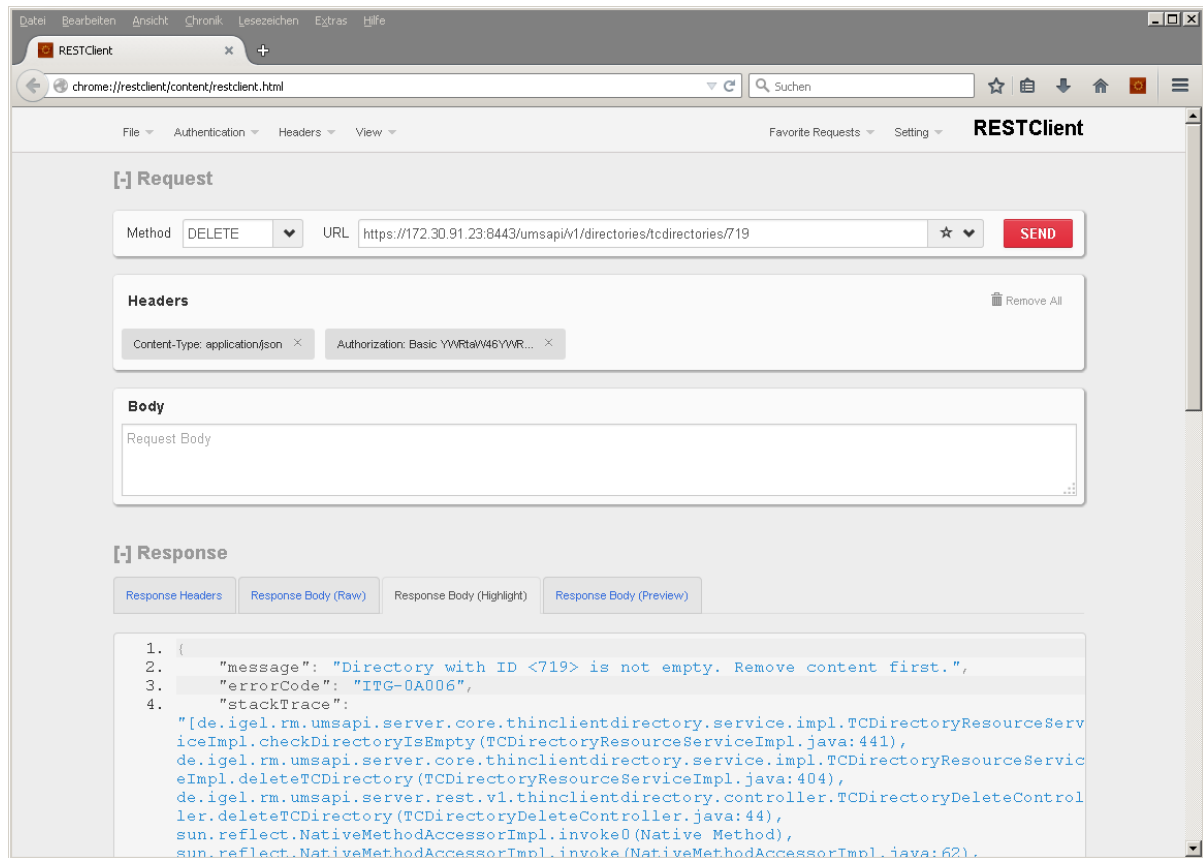
The easiest way to try out the *IGEL Management Interface* is either

- with [RESTClient](#)⁷, an add-on for *Mozilla Firefox*:

⁵ <https://kb.igel.com/pages/viewpage.action?pagelId=47711318>

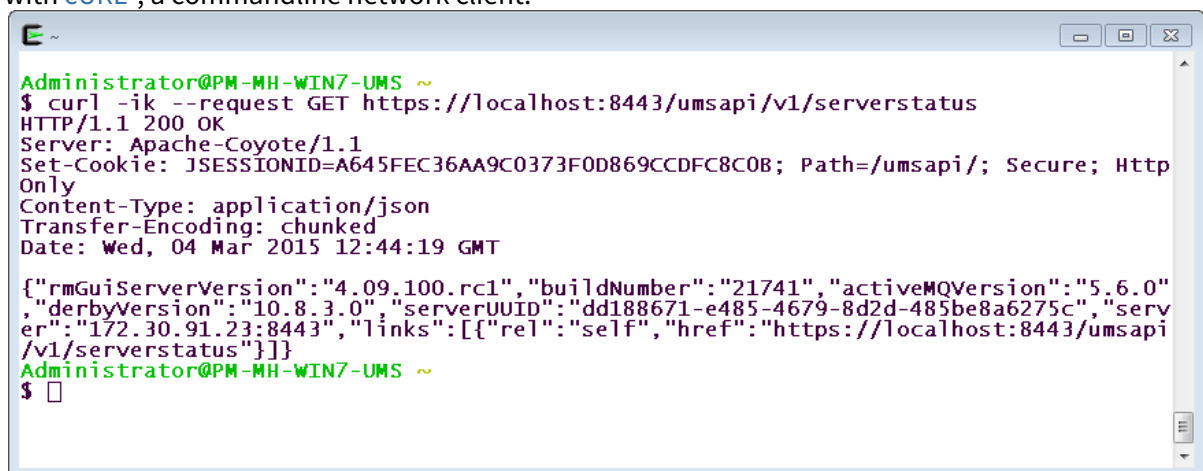
⁶ <https://kb.igel.com/pages/viewpage.action?pagelId=43106291>

⁷ <http://restclient.net/>



or

- with `cURL`⁸, a commandline network client:



This guide uses `cURL` for examples, as the commandline makes all parameters visible in plain text. This need not stop you from using RESTClient if you are more comfortable with it. You can easily translate the commandline parameters into the fields of the RESTClient *RESTClient* GUI.

⁸ <http://curl.haxx.se/>



Programming libraries

Most programming languages provide an HTTP and an SSL module, either in their standard library or as an extension, and a JSON library (for the API data format) as well.

1.4.2 Getting the Server Status

- Send the endpoint `/v3/serverstatus` an HTTP GET request to retrieve some information about the server.

With *cURL* the commandline looks like this:

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/serverstatus
```

You will receive a response like the following:

```
200 OK
---
{
  "rmGuiServerVersion": "5.07.100.rc15",
  "buildNumber": "32287",
  "activeMQVersion": "5.6.0",
  "derbyVersion": "10.12.1.1",
  "serverUUID": "9fe719f1-c16e-4744-9ff1-b9c314ae151c",
  "server": "mhuber:8443",
  "links": []
}
```

Response Headers

The server replied with the status code 200 in the HTTP header, which means that your request was successful.

You can find a list of status codes in the [IMI API Reference](#)(see page 63).

JSON Response Body

Information about the server is given in the response body in the *JavaScript Object Notation (JSON)* format. It is enclosed in curly braces and contains key-value-pairs separated by a colon. The line separator is a comma. Your programming language probably offers a JSON module that will help you parse and process information in this format.



1.4.3 Authentication

You must be authenticated in order to use *IGEL Management Interface*, otherwise the server will return the HTTP status 401 "Unauthorized". Only querying the server status is allowed without authentication.

IMI uses *Basic Authentication* ([RFC 2617](#)(see page 18)).

Logging In

- Send an HTTP POST request to `/v2/login` to log in. Your HTTP client of choice will offer a way of sending a *Basic Authentication*- header, which is produced from the username/password combination.

This is the commandline for *cURL*:

```
curl \
--request POST \
--user '[Username]:[Passwort]' \
https://[server]:8443/umsapi/v3/login
```

The server replies, sending back a session ID:

```
200 OK
Set-Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC; Path=/umsapi/; Secure; HttpOnly
---
{"message":"JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC"}
```

Staying Logged in

- You can maintain the session by sending the JSESSIONID in the Cookie header with every subsequent request:

```
Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC
```

Some clients will do this automatically for you, e.g. the [RESTClient](#) (see page 15) [Firefox add-on](#) (see page 15).

1.4.4 Listing all Thin Clients

- Send a GET request to `/v3/thinclients` to list all thin clients.
- Don't forget to tell the server your JSESSIONID:

```
curl \
--request GET \
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
https://[server]:8443/umsapi/v3/thinclients
```



Response

The response contains a large *JSON* document, listing all thin clients and their most important properties. The links make further parts of the API discoverable for a client that spiders the interface.

Look up the format for a thin client resource in the [IMI API Reference](#) (see page 63).

The last thin client entry contains "movedToBin": true, meaning that the thin client has been deleted and moved to the **Recycling Bin**. Objects in the **Recycling Bin** are listed, but you cannot update them or send them commands.

```
[
{
  "unitID": "00E0C54DCB8E",
  "mac": "00E0C54DCB8E",
  "firmwareID": "21",
  "lastIP": "172.30.91.43",
  "id": "27",
  "name": "Front Desk",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tc",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/27"
    },
    {
      "rel": "Parent",
      "href": "root"
    },
    {
      "rel": "Firmware",
      "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/21"
    }
  ]
},
{

```



```

"unitID": "DC9C5207694E",
"mac": "DC9C5207694E",
"firmwareID": "13",
"lastIP": "172.30.91.24",
"id": "6888",
"name": "UD3 M340C_Board",
"parentID": "15592",
"movedToBin": false,
"objectType": "tc",
"links": [...]
},
{
"unitID": "00E0C5080834",
"mac": "00E0C5080834",
"firmwareID": "2",
"lastIP": "172.30.91.132",
"id": "6899",
"name": "UD10",
"parentID": "15592",
"movedToBin": true,
"objectType": "tc",
"links": [...]
}
]

```

1.4.5 Getting Information on a Thin Client

- Send a GET request to `/v3/thinclients/[id]` to get information about a thin client:

```

curl \
--request GET\
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
https://[server]:8443/umsapi/v3/thinclients/27

```

The thin client ID is passed to the API as a *Path Variable* in the URL.

The response contains the most important properties of the thin client with the ID 27:



Response

```
{
  "unitID": "00E0C54DCB8E",
  "mac": "00E0C54DCB8E",
  "firmwareID": "21",
  "lastIP": "172.30.91.43",
  "id": "27",
  "name": "Front Desk",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tc",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/27"
    },
    {
      "rel": "Parent",
      "href": "root"
    },
    {
      "rel": "Firmware",
      "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/21"
    }
  ]
}
```

1.4.6 Getting All Details about a Thin Client

- Use the details facet in order to get all details about a thin client:

```
curl \
--request GET\
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
https://[server]:8443/umsapi/v3/thinclients/27?facets=details
```



The response contains all the properties of the thin client with the ID 27:

Response

```
{
  "unitID": "00E0C54DCB8E",
  "mac": "00E0C54DCB8E",
  "firmwareID": "21",
  "networkName": "D.Weinert",
  "site": "1. Stock",
  "department": "Product Management",
  "lastIP": "172.30.91.43",
  "costCenter": "",
  "comment": "",
  "assetID": "",
  "inServiceDate": "01.06.215",
  "serialNumber": "",
  "productId": "UD6-LX 51cps",
  "umsStructuralTag": "",
  "cpuSpeed": 2416,
  "cpuType": "Intel(R) Celeron(R) CPU J1900 @ 1.99GHz",
  "deviceType": "IGEL H830C",
  "deviceSerialNumber": "14D3D3C03B14470B9EM",
  "osType": "IGEL Linux V5 (Kernel Version 3.13.11-ckt20)",
  "flashSize": 1883,
  "memorySize": 1853,
  "networkSpeed": 1000,
  "graphicsChipset0": "INTEL HD Graphics (Baytrail)",
  "graphicsChipset1": "",
  "monitorVendor1": "Samsung Electric Company",
  "monitorModel1": "S24C650",
  "monitorSerialnumber1": "H4MG404381",
  "monitorSize1": 24,
  "monitorNativeResolution1": "1920 x 1200",
  "monitor1YearOfManufacture": "2015",
```



```
"monitor1WeekOfManufacture": "17",
"monitorVendor2": "Samsung Electric Company",
"monitorModel2": "S24C650",
"monitorSerialnumber2": "H4MG404389",
"monitorSize2": 24,
"monitorNativeResolution2": "1920 x 1200",
"monitor2YearOfManufacture": "2015",
"monitor2WeekOfManufacture": "17",
"biosVendor": "INSYDE Corp.",
"biosVersion": "H830C V:3.5.13-11282014",
"biosDate": "11/28/2014",
"totalUsagetime": "5798368000",
"totalUptime": "18513000",
"lastBoottime": "2015-09-29 08:30",
"id": "27",
"name": "Front Desk",
"parentID": "-1",
"movedToBin": false,
"objectType": "tc",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/thinclients/27"
},
{
"rel": "Parent",
"href": "root"
},
{
"rel": "Firmware",
"href": "https://172.30.91.227:8443/umsapi/v3/firmwares/21"
}
]
```



1.4.7 Getting Thin Client Status

- Use the `online` facet in order to get the online status of a thin client with a specific ID:

```
curl \
--request GET\
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
https://[server]:8443/umsapi/v3/thinclients/27?facets=online
```

The response contains, among others, the `online` property, which can have the values `true` and `false`:

Response

```
{
  "unitID": "00E0C54DCB8E",
  "mac": "00E0C54DCB8E",
  "firmwareID": "21",
  "lastIP": "172.30.91.43",
  "online": false,
  "id": "27",
  "name": "Front Desk",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tc",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/27"
    },
    {
      "rel": "Parent",
      "href": "root"
    },
    {
      "rel": "Firmware",
      "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/21"
    }
  ]
}
```




```
}
]
}
```

1.5 Creating, Updating and Deleting Resources

The thin client directories in UMS are a good resource for trying out the creation, update and deletion API calls.

-
- [Listing all Thin Client Directories](#)(see page 25)
 - [Creating a Thin Client Directory](#)(see page 27)
 - [Updating a Thin Client Directory](#)(see page 28)
 - [Deleting a Thin Client Directory](#)(see page 29)

1.5.1 Listing all Thin Client Directories

- List all thin client directories:

```
curl \
--request GET\
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
https://[server]:8443/umsapi/v3/directories/tcdirectories/
```

Response

The response contains a list of thin client directories in *JSON* format. The ID of the root thin client directory is always -1. Find a detailed description of the **Thin Client Directory** resource in the [IMI API Reference](#)(see page 63).

- Compare the *JSON* response to the thin client directory tree you see in your *UMS console*.



IMI provides a flat view of these directories, subdirectories can only be recognized via their parentID. In order to better see the nesting, use the `children` facet.



```
[
{
  "id": "15592",
  "name": "Pool",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tcdirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/15592"
    },
    {
      "rel": "Parent",
      "href": "root"
    }
  ]
},
{
  "id": "76863",
  "name": "New Subdirectory",
  "parentID": "76462",
  "movedToBin": false,
  "objectType": "tcdirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76863"
    },
    {
      "rel": "Parent",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76462"
    }
  ]
}
```



```

]
},
{
  "id": "76462",
  "name": "New Directory",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tcdirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76462"
    },
    {
      "rel": "Parent",
      "href": "root"
    }
  ]
}
]

```

1.5.2 Creating a Thin Client Directory

► Send an HTTP PUT request to the resource `/v3/directories/tcdirectories` to create a new thin client directory.

This request must contain *JSON* data in its body specifying a name for the new directory. The `parentID` is optional and defaults to the root thin client directory:

```

{
  "name": "My Directory",
  "parentID": "-1"
}

```

cURL accepts *JSON* data as the `--data` parameter. This request also specifies `Content-type` header to tell *IMI* that the data has the type `application/json`.

► Use it whenever you send *JSON* data:

```
curl \
```



```
--request PUT \
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
--header "Content-type: application/json" \
--data '{"name":"My Directory", "parentID":"-1"}' \
https://[server]:8443/umsapi/v3/directories/tcdirectories
```

Response

IMI replies with a success message and data about the newly created directory:

```
{
  "message": "Directory successfully inserted.",
  "id": "77118",
  "name": "My Directory",
  "parentID": "-1"
}
```

1.5.3 Updating a Thin Client Directory

► Send an HTTP PUT request to `/v3/directories/tcdirectories/[id]` to update the properties of a thin client directory. The directory ID is passed as a variable in the URL, a new directory name as *JSON* data in the request body. Do not forget the `Content-type` header.

```
curl \
--request PUT \
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
--header "Content-type: application/json" \
--data '{"name":"My Wonderful Directory"}' \
https://[server]:8443/umsapi/v3/directories/tcdirectories/77118
```

Response

► *IMI* replies with a success message:

```
200 OK
---
{
  "message": "Updated directory successfully."
}
```



1.5.4 Deleting a Thin Client Directory

► Send an HTTP DELETE request to `/v3/directories/tcdirectories/[id]` to delete a thin client directory. The ID of the directory is passed as a path variable in the URL.

```
curl \
--request DELETE \
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
https://[server]:8443/umsapi/v3/directories/tcdirectories/77118
```

Response

► IMI replies with a brief success message:

Code Example:

```
<code> 200 OK</code>
```

Code Example:

```
<code>---</code>
```

Code Example:

```
<code> {"message":"Deletion successful."} </code>
```

A thin client directory has to be empty to be removed. If you try to delete a non-empty directory, IMI will respond with an error message:

```
400 Bad Request
```

```
---
```

```
{
  "message": "Directory with ID <77118> is not empty. Remove content first.",
  "errorCode": "ITG-0A006",
  "time": "2016-04-06T13:29:59.362",
  "stackTrace": "[de.igel.rm.umsapi.server.services.thinclientdirectory.
[...]]
}
```

Learn more about error codes and messages in the [IMI API Reference](#)(see page 63).



1.6 Further Operations

- [Moving a Thin Client Directory](#)(see page 30)
- [Sending a Command to Thin Clients](#)(see page 31)
- [Debugging Requests](#)(see page 32)

1.6.1 Moving a Thin Client Directory

► In order to move thin clients or directories send a PUT request to the target directory and append ? operation=move. The request body must contain a list of API Objects in *JSON* format, representing the thin clients and directories to be moved. An API Object has an ID and a type.

```
curl \
--request PUT \
--header 'Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
--header "Content-type: application/json" \
--data '[ { "id":"77123", "type":"tcdirectory"},\
{ "id":"1234", "type":"tcdirectory" } ]' \
https://[server]:8443/umsapi/v3/directories/tcdirectories/15592?operation=move
```

Response

The following response contains an error message for one directory and a success message for the other:

```
200 OK
---
[
{
  "id": "1234",
  "results": "does_not_exist"
},
{
  "id": "77123",
  "results": "successful"
}
]
```



1.6.2 Sending a Command to Thin Clients

► Send a POST request to `/v3/thinclients?command=[command]` to send a command to one or more thin clients. A command can be one of the following:

- reboot
- shutdown
- wakeup
- settings2tc
- tcreset2facdefs

The request body contains a list of API Objects representing the thin clients addressed.

To send the reboot command to two thin clients, send the following request to *IMI*:

```
curl \
--request POST \
JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC' \
--header "Content-type: application/json" \
--data '[{"id":"27", "type":"tc"}, {"id":"72014", "type":"tc"}]' \
https://[server]:8443/umsapi/v3/thinclients?command=reboot
```

Request

IMI replies with a *JSON* document containing a result for each thin client addressed:

```
{
  "CommandExecList": [
    {
      "execID": "ID-PM-MH-WIN7-UMS-54530-1456839861871-5-0",
      "id": "72014",
      "mac": "00E0C561EEED",
      "exectime": "1456845240566",
      "message": "OK",
      "state": "SUCCESS"
    },
    {
      "execID": "ID-PM-MH-WIN7-UMS-54530-1456839861871-5-0",
      "id": "27",
      "mac": "00E0C54DCB8E",
```



```
"exectime": "1456845240560",  
"message": "OK",  
"state": "SUCCESS"  
}  
]  
}
```

1.6.3 Debugging Requests

Error Codes

To debug errors when communicating with IMI, observe

- the HTTP status in the response headers
- the messages and error codes in the response bodies.

HTTP-Proxy

You may use an HTTP proxy in order to see what requests your scripts or programs produce. A proxy acts as a man-in-the-middle between client and IMI, letting you view and optionally edit requests.

An Open Source program for this task is [OWASP ZAP](https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project)⁹.

Use a proxy only for debugging your own HTTP requests. Viewing other users' network traffic is usually illegal.

⁹ https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project



Untitled Session - OWASP ZAP 2.4.1

File Edit View Analyse Report Tools Online Help

Standard mode

Sites + Quick Start Request Response +

Contexts

- Default Context
- Sites
 - https://localhost:8443

Header: Text Body: Text

```
GET https://localhost:8443/umsapi/v1/thinclients HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: null
Authorization: Basic YWRtaW46YWRtaW4=
Content-Type: application/json
Cookie: JSESSIONID=39C0C727F655917DBD7B72254E1FF04A
Connection: keep-alive
Host: localhost:8443
```

History Search Alerts Output +

Filter: OFF

Id	Req. Timesta...	Meth...	URL	Co...	Reason	R...	Size	Resp. B...	Highest A...	N...	Tags
1	23/09/15 10:3...	GET	https://localhost:8443/umsapi/v1/thind...	200	OK	3...	5.58 KiB		Medium		
5	23/09/15 10:3...	PUT	https://localhost:8443/umsapi/v1/thind...	400	Bad Re...	6...	3.63 KiB		Medium		

Alerts 0 0 1 4 0

Current Scans 0 0 0 0 0 0 0 0



2 Powershell

- [Introduction](#)(see page 34)
- [Command Reference](#)(see page 36)

2.1 Introduction

IMI Cmdlets is a collection of Windows PowerShell commandlets for use with the *IGEL Management Interface (IMI)*. The commandlets are installed as binary *.dll files and can be used in *PowerShell* scripts. They simplify frequent actions such as rebooting a thin client or assigning a profile via *IMI*, hiding the implementation details of the underlying REST API.

Microsoft .NET version 4.5 or greater is required for IMI Cmdlets 1.04.100 for TLSv1.2

-
- [Installation](#)(see page 34)
 - [Loading the IMI Cmdlets Snap-in](#)(see page 35)
 - [Authentication](#)(see page 35)

2.1.1 Installation

Prerequisites

- One of the following *Microsoft* operating systems:
 - *Windows 7*
 - *Windows Server 2008 or 2008 R2*
 - *Windows 8 or higher*
 - *Windows Server 2012 or 2012 R2*
- *Windows Management Framework 4* (which already includes *PowerShell 4*)
- *Microsoft .Net Framework 4* or newer

IGEL Management Interface (IMI) PowerShell Cmdlets 1.03.100 or lower do not support UMS 5.08.100 or higher.
UMS 5.08.100 or higher uses TLS 1.2, which is supported as of PowerShell Cmdlets 1.04.100 only.

Installing

1. Download the *IMI* Commandlets Installer from the [download server](#)¹⁰.
2. Launch the installer.

¹⁰ <https://www.igel.com/software-downloads/>



You need administration privileges in order to install IMI Commandlets.

3. Close all other applications and confirm that you have done so.
 4. Review and accept the **License Agreement**.
 5. Select the destination location for the installation or leave the default.
 6. Optional: Select a Start Menu folder for the program shortcuts.
 7. Read the summary and start the installation process.
- The installer will install the IMI Commandlets and some sample scripts. The following shortcuts will be placed on the desktop:
- **Starter-Interface:** A simple interactive terminal interface to *IMI*, suitable for new users
 - **IGEL-Shell:** A *Windows* PowerShell session with *IMI* Commandlets already loaded

Additionally, the *IMI* Commandlets Readme file is displayed.

2.1.2 Loading the IMI Cmdlets Snap-in

Before you can use IMI Cmdlets, you need to add the snap-in to your PowerShell session:

```
Add-PSSnapin *igel*
```

View all loaded IGEL Cmdlets:

```
Get-PSSnapin *igel*
```

2.1.3 Authentication

IMI uses HTTP Basic Authentication with a username-password combination. IMI Cmdlets wrap authentication in the **Get-IgelRMServerLogin** cmdlet.

Get-IgelRMServerLogin logs in to IMI.

Parameters

- **-Servername**
Hostname of the of the UMS server with IMI
- **-Username**
UMS user name
- **-ForceIMIVersion**
Use a specific IMI version, v1 or v2.
- **-IgnoreUntrustedCertificates**
If this is set to \$true the Cmdlets will ignore errors in TLS/SSL certificate validation.

The password is entered interactively.



Example Command Line

Log in and save the credentials for the session in \$l:

```
$l = Get-IgelRMServerLogin `
  -Servername localhost `
  -Username SPIELWIESE `
  -IgnoreUntrustedCertificates $true `
  -ForceIMIVersion v2
```

Supply the credentials to subsequent commands using

`-Credentials $l`

2.2 Command Reference

- [Thin Client](#)(see page 36)
- [Profile](#)(see page 41)
- [Master Profile](#)(see page 45)
- [Thin Client Directory](#)(see page 50)
- [Profile Directory](#)(see page 54)
- [Master Profile Directory](#)(see page 58)
- [Firmware](#)(see page 61)
- [Server Status](#)(see page 62)

2.2.1 Thin Client

- [Get-IgelTCInformation](#)(see page 36)
- [Start-IgelTCs](#)(see page 38)
- [Stop-IgelTC](#)(see page 38)
- [Restart-IgelTCs](#)(see page 39)
- [Move-IgelTCs](#)(see page 39)
- [Update-IgelTCSettings](#)(see page 40)

Get-IgelTCInformation

Summary

Gets information on thin clients.



Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login.
- **-tcID**
Numeric ID of the thin client. If omitted, information is retrieved on all thin clients, including those in the Recycling Bin.
- **-Details**
Level of detail in the output, one of:
 - **full** Gets all data available from IMI, including the online status.
 - **inventory** Gets all data available from IMI, apart from the online status.
 - **online** Gets data in a short format, including the online status.

If this parameter is omitted, only a short information format with the most important thin client properties is returned.

Example Command Line

```
Get-IgelTCInformation -Credentials $l
```

Example Output

```
mac : 000BCA050027
firmwareID : 45
lastIP : 172.30.91.237
id : 48335
name : UD2-D220
parentID : 76863
movedToBin : True
objectType : tc

mac : 005056934FDB
firmwareID : 28
lastIP : 172.30.91.9
id : 48366
name : westestx86
parentID : -1
movedToBin : False
objectType : tc

mac : 00E0C561EEED
```



```
firmwareID : 43
lastIP : 172.30.91.30
id : 72014
name : IGEL-8KJ2GQPL3N
parentID : 76863
movedToBin : False
objectType : tc
```

Start-IgelTCs

Summary

Wakes up thin clients.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-tcID / -tcIDs**
Numeric IDs of the thin clients to wake up

Example Command Line

```
Start-IgelTCs -Credentials $l -tcIDs 27
```

Example Output

```
execID exectime message state
-----
ID-PM-MH-WIN7-UMS-49242-14... 1460647127627 OK SUCCESS
```

Stop-IgelTC

Summary

Shuts down thin clients.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-tcID / -tcIDs**
Numeric IDs of the thin clients to shut down



Example Command Line

```
Stop-IgelTCs -Credentials $l -tcID 2010
```

Example Output

```
execID exectime message state
```

```
-----
```

```
ID-mhuber-50400-14605614057... 1460706641670 OK SUCCESS
```

Restart-IgelTCs

Summary

Reboots thin clients.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-tcIDs**
Numeric IDs of the thin clients to reboot

Example Command Line

```
Restart-IgelTCs -Credentials $l -tcIDs 2010,1435
```

Example Output

```
execID exectime message state
```

```
-----
```

```
ID-mhuber-50400-14605614057... 1460991770313 OK SUCCESS
```

```
ID-mhuber-50400-14605614057... 1460991770313 OK SUCCESS
```

Move-IgelTCs

Summary

Moves a thin client to a thin client directory.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login



- **-tcIDs**
Numeric ID of the thin client
- **-DirectoryID**
Numeric ID of the target thin client directory

Example Command Line

```
Move-IgelTCs `
-Credentials $l `
-TCIDs 2010 `
-DirectoryID 3147
```

Example Output

```
id results
--
2010 successful
```

Update-IgelTCSettings

Summary

Sends settings from UMS to the thin client.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-tcID**
Numeric ID of the thin client

Example Command Line

```
Update-IgelTCSettings -Credentials $l -tcID 2010
```

Example Output

```
execID : ID-mhuber-50400-1460561405782-16-0
id : 2010
mac : 000BCA050027
exectime : 1460708102541
message : OK
state : SUCCESS
```




2.2.2 Profile

- [Get-IgelTCProfile](#)(see page 41)
- [Rename-IgelTCProfile](#)(see page 41)
- [Get-IgelProfileAssignments](#)(see page 42)
- [Set-IgelProfileAssignment](#)(see page 43)
- [Remove-IgelProfileAssignment](#)(see page 44)
- [Remove-IgelTCProfile](#)(see page 45)

Get-IgelTCProfile

Summary

Gets information about profiles.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile. If this is omitted, information is retrieved for all profiles

Example Command Line

```
Get-IgelTCProfile -Credentials $l -ProfileID 3150
```

Example Output

```
firmwareID : 3
isMasterProfile : False
overridesSessions : False
id : 3150
name : My Profile
parentID : -2
movedToBin : False
objectType : profile
```

Rename-IgelTCProfile

Summary

Renames a profile.



Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile
- **-newProfileName**
New profile name as string

Example Command Line

```
Rename-IgelTCProfile `
-Credentials $l `
-ProfileID 3150 `
-newProfileName "Updated Profile"
```

Example Output

message

Update successful

Get-IgelProfileAssignments

Summary

Gets profile assignments.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile
- **-ObjectType**
Get assignments to a specific object type, one of:
 - **tc** thin client
 - **directory** thin client directory

Example Command Line

```
Get-IgelProfileAssignments `
-Credentials $l `
-ProfileID 3150 `
```



`-ObjectType tc`

Example Output

```
type id assignmentPosition
---- -- -----
tc 2010 0
```

Set-IgelProfileAssignment

Summary

Creates a profile or master profile assignment.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile or master profile
- **-ProfileType**
Profile type, one of:
 - **profile** profile
 - **masterprofile** master profile
- **-TargetID**
Numeric ID of the target object which the profile or master profile will be assigned to
- **-TargetType**
Assignments to a specific object type, one of:
 - **tc** thin client
 - **tcdirectory** thin client directory

Example Command Line

```
Set-IgelProfileAssignment `
-Credentials $l `
-ProfileID 3150 `
-ProfileType profile `
-TargetID 2010 `
-TargetType tc
```

Example Output

```
message
-----
```



1 assignments successfully assigned

Remove-IgelProfileAssignment

Summary

Deletes a profile or master profile assignment.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile or master profile
- **-ProfileType**
Profile type, one of:
 - **profile** profile
 - **masterprofile** master profile
- **-TargetID**
Numeric ID of the target object which the profile or master profile is assigned to
- **-TargetType**
Assignments to a specific object type, one of:
 - **tc** thin client
 - **tcdirectory** thin client directory

Example Command Line

```
Remove-IgelProfileAssignment `
-Credentials $l `
-ProfileID 3151 `
-ProfileType profile `
-TargetType tcdirectory `
-TargetID 3201
```

Example Output

message

deleted profile assignment



Remove-IgelTCProfile

Summary

Deletes a profile.

This does not move the profile to the Recycling Bin, but simply deletes it.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile

Example Command Line

```
Remove-IgelProfile -Credentials $l -ProfileID 3150
```

Example Output

message

Deleted profile with id 3150

2.2.3 Master Profile

- [Get-IgelTCMasterprofile](#)(see page 45)
- [Rename-IgelTCMasterprofile](#)(see page 46)
- [Get-IgelMasterprofileAssignments](#)(see page 47)
- [Set-IgelProfileAssignment](#)(see page 47)
- [Remove-IgelProfileAssignment](#)(see page 48)
- [Remove-IgelMasterprofile](#)(see page 49)

Get-IgelTCMasterprofile

Summary

Gets information about master profiles.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login



- **-MasterprofileID**

Numeric ID of the master profile. If this is omitted, information is retrieved for all master profiles

Example Command Line

```
Get-IgelTCMasterprofile -Credentials $l -MasterprofileID 3152
```

Example Output

```
firmwareID : 3
isMasterProfile : True
overridesSessions : False
id : 3152
name : My Master Profile
parentID : -14
movedToBin : False
objectType : masterprofile
```

Rename-IgelTCMasterprofile

Summary

Renames a master profile.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-MasterprofileID**
Numeric ID of the master profile
- **-newMasterProfileName**
New master profile name as string

Example Command Line

```
Rename-IgelTCMasterprofile `
-Credentials $l `
-MasterprofileID 3152 `
-newMasterProfileName "Renamed Master Profile"
```

Example Output

```
message
```



Update successful

Get-IgelMasterprofileAssignments

Summary

Gets master profile assignments.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-MasterprofileID**
Numeric ID of the master profile
- **-ObjectType**
Get assignments to a specific object type, one of:
 - **tc** thin client
 - **tcdirectory** thin client directory

Example Command Line

```
Get-IgelMasterprofileAssignments `
-Credentials $l `
-MasterprofileID 3152 `
-ObjectType tc
```

Example Output

```
type id assignmentPosition
---- --
tc 2010 0
```

Set-IgelProfileAssignment

Summary

Creates a profile or master profile assignment.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login



- **-ProfileID**
Numeric ID of the profile or master profile
- **-ProfileType**
Profile type, one of:
 - **profile** profile
 - **masterprofile** master profile
- **-TargetID**
Numeric ID of the target object which the profile or master profile will be assigned to
- **-TargetType**
Assignments to a specific object type, one of:
 - **tc** thin client
 - **tcdirectory** thin client directory

Example Command Line

```
Set-IgelProfileAssignment `
-Credentials $l `
-ProfileID 3150 `
-ProfileType profile `
-TargetID 2010 `
-TargetType tc
```

Example Output

```
message
-----
1 assignments successfully assigned
```

Remove-IgelProfileAssignment

Summary

Deletes a profile or master profile assignment.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile or master profile
- **-ProfileType**
Profile type, one of:
 - **profile** profile
 - **masterprofile** master profile



- **-TargetID**
Numeric ID of the target object which the profile or master profile is assigned to
- **-TargetType**
Assignments to a specific object type, one of:
 - **tc** thin client
 - **tcdirectory** thin client directory

Example Command Line

```
Remove-IgelProfileAssignment `
-Credentials $l `
-ProfileID 3151 `
-ProfileType profile `
-TargetType tcdirectory `
-TargetID 3201
```

Example Output

```
message
-----
deleted profile assignment
```

Remove-IgelMasterprofile

Summary

Deletes a master profile.

This does not move the master profile to the Recycling Bin, but simply deletes it.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-MasterprofileID**
Numeric ID of the master profile

Example Command Line

```
Remove-IgelMasterprofile -Credentials $l -MasterprofileID 3152
```

Example Output

```
message
```



Deleted profile with id 3152

2.2.4 Thin Client Directory

- [Get-IgelTCDirectories](#)(see page 50)
- [Rename-IgelTCDirectory](#)(see page 51)
- [Set-IgelTCDirectory](#)(see page 52)
- [Move-IgelTCDirectory](#)(see page 52)
- [Set-IgelProfileAssignment](#)(see page 53)
- [Remove-IgelTCDirectory](#)(see page 54)

Get-IgelTCDirectories

Summary

Gets information on thin client directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-DirectoryID**
Numeric ID of the thin client directory. If omitted, information is retrieved on all thin client directories, including those in the Recycling Bin
- **-details**
Amount of detail in the output:
 - **\$true** The IDs of `DirectoryChildren` will be shown, which can be thin clients or thin client directories.

Example Command Line

```
Get-IgelTCDirectories -Credentials $l -details $true
```

Example Output

```
id : 3201
name : B Directory
parentID : -1
movedToBin : False
objectType : tcdirectory
DirectoryChildren : {3202, 1435}
```



```
id : 3202
name : Subdirectory
parentID : 3201
movedToBin : False
objectType : tcdirectory
DirectoryChildren : {}
```

```
id : 3147
name : A Directory
parentID : -1
movedToBin : False
objectType : tcdirectory
DirectoryChildren : {2010}
```

Rename-IgelTCDirectory

Summary

Renames a thin client directory.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryID**
Numeric ID of the thin client directory
- **-newDirectoryName**
New thin client directory name as string

Example Command Line

```
Rename-IgelTCDirectory `
-Credentials $l `
-directoryID 3206 `
-newDirectoryName "Renamed TC Directory"
```

Example Output

```
message
```

```
-----
```

```
Updated directory successfully.
```



Set-IgelTCDirectory

Summary

Creates a new thin client directory.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryName**
Name for the thin client directory
- **-directoryPosition**
Parent directory for the new thin client directory. If omitted, it defaults to -1, which is the UMS directory **Thin Clients**.

Example Command Line

```
Set-IgelTCDirectory `
-Credentials $l `
-directoryName "Brand New Directory" `
-directoryPosition 3201
```

Example Output

```
message id name parentID
----- -- ----
Directory successfully inse... 3206 Brand New Directory 3201
```

Move-IgelTCDirectory

Summary

Moves one or more thin client directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryIDs**
Numerical IDs of the thin client directories to be moved
- **-targetDirectory**
The target directory to move the thin client directories into



Example Command Line

```
Move-IgelTCDirectory `
-Credentials $l `
-directoryIDs 3206 `
-targetDirectory
```

Example Output

```
id results
--
3206 successful
```

Set-IgelProfileAssignment

Summary

Creates a profile or master profile assignment.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-ProfileID**
Numeric ID of the profile or master profile
- **-ProfileType**
Profile type, one of:
 - **profile** profile
 - **masterprofile** master profile
- **-TargetID**
Numeric ID of the target object which the profile or master profile will be assigned to
- **-TargetType**
Assignments to a specific object type, one of:
 - **tc** thin client
 - **tcdirectory** thin client directory

Example Command Line

```
Set-IgelProfileAssignment `
-Credentials $l `
-ProfileID 3150 `
-ProfileType profile `
-TargetID 2010 `
```



`-TargetType tc`

Example Output

message

1 assignments successfully assigned

Remove-IgelTCDirectory

Summary

Deletes a thin client directory.

You can only delete empty thin client directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryID**
Numerical ID of the thin client directory to be deleted

Example Command Line

```
Remove-IgelTCDirectory -Credentials $l -directoryID 3202
```

Example Output

message

Deletion successful.

2.2.5 Profile Directory

- [Get-IgelProfileDirectory](#)(see page 55)
- [Rename-IgelTCProfileDirectory](#)(see page 55)
- [Move-IgelTCProfileDirectory](#)(see page 56)
- [Set-IgelProfileDirectory](#)(see page 56)
- [Remove-IgelProfileDirectory](#)(see page 57)



Get-IgelProfileDirectory

Summary

Gets information on profile directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-DirectoryID**
Numeric ID of the profile directory. If omitted, information is retrieved on all profile directories, including those in the Recycling Bin.

Example Command Line

```
Get-IgelProfileDirectory -Credentials $l -ProfileDirectoryID 3221
```

Example Output

```
id : 3221
name : Profiles 2
parentID : -2
movedToBin : False
objectType : profiledirectory
```

Rename-IgelTCProfileDirectory

Summary

Renames a profile directory.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryID**
Numeric ID of the profile directory
- **-newDirectoryName**
New profile directory name as string

Example Command Line

```
Rename-IGELTCProfileDirectory `
-Credentials $l `
```



```
-directoryID 3220 `
-newDirectoryName "My Renamed Directory"
```

Example Output

message

Updated directory successfully.

Move-IgelTCProfileDirectory

Summary

Moves one or more profile directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryIDs**
Numerical IDs of the profile directories to be moved
- **-targetDirectory**
The target directory to move the profiles directories into

Example Command Line

```
Move-IgelProfileDirectory `
-Credentials $l `
-directoryIDs 4024 `
-targetDirectory 3220
```

Example Output

id results

-- -----

4024 successful

Set-IgelProfileDirectory

Summary

Creates a new profile directory.



Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryName**
Name for the profile directory
- **-directoryPosition**
Parent directory for the new profile directory. If omitted, it defaults to -2, which is the UMS directory **Profiles**.

Example Command Line

```
Set-IgelProfileDirectory `
-Credentials $l `
-directoryName "A New Directory" `
-directoryPosition -2
```

Example Output

```
message id name parentID
```

```
----- -- ----
```

```
Directory successfully inse... 4030 A New Directory -2
```

Remove-IgelProfileDirectory

Summary

Deletes a profile directory.

You can only delete empty thin client directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryID**
Numerical ID of the profile directory to be deleted

Example Command Line

```
Remove-IgelProfileDirectory -Credentials $l -ProfileDirectoryID 4024
```



Example Output

message

Deletion successful.

2.2.6 Master Profile Directory

- [Get-IgelMasterprofileDirectory](#)(see page 58)
- [Rename-IgelTCMasterprofileDirectory](#)(see page 59)
- [Move-IgelMasterprofileDirectory](#)(see page 59)
- [Set-IgelMasterprofileDirectory](#)(see page 60)
- [Remove-IgelMasterprofileDirectory](#)(see page 60)

Get-IgelMasterprofileDirectory

Summary

Gets information on master profile directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-DirectoryID**
Numeric ID of the master profile directory. If omitted, information is retrieved on all master profile directories, including those in the Recycling Bin

Example Command Line

```
Get-IgelMasterprofileDirectory -Credentials $l -MasterprofileDirectoryID 4034
```

Example Output

```
id : 4034
name : New Master Profiles
parentID : -14
movedToBin : False
objectType : masterprofiledirectory
```



Rename-IgelTCMasterprofileDirectory

Summary

Renames a master profile directory.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryID**
Numeric ID of the master profile directory
- **-newDirectoryName**
New master profile directory name as string

Example Command Line

```
Rename-IGELTCMasterProfileDirectory `
-Credentials $l `
-directoryID 4038 `
-newDirectoryName "Renamed Master Profiles"
```

Example Output

message

Updated directory successfully.

Move-IgelMasterprofileDirectory

Summary

Moves one or more master profile directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryIDs**
Numerical IDs of the master profile directories to be moved
- **-targetDirectory**
The target directory to move the master profiles directories into



Example Command Line

```
Move-IgelMasterprofileDirectory `
-Credentials $l `
-directoryIDs 4033 `
-targetDirectory 4034
```

Example Output

```
id results
-- -----
4033 successful
```

Set-IgelMasterprofileDirectory

Summary

Creates a new master profile directory.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryName**
Name for the master profile directory
- **-directoryPosition**
Parent directory for the new master profile directory. If omitted, it defaults to -14, which is the UMS directory **Master Profiles**.

Example Command Line

```
Set-IgelMasterprofileDirectory -Credentials $l -directoryName "Brand New Directory"
```

Example Output

```
message id name parentID
----- -- -
Directory successfully inse... 4038 Brand New Directory -14
```

Remove-IgelMasterprofileDirectory

Summary

Deletes a master profile directory.



You can only delete empty directories.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-directoryID**
Numerical ID of the master profile directory to be deleted

Example Command Line

```
Remove-IgelMasterprofileDirectory -Credentials $l -MasterprofileDirectoryID 4033
```

Example Output

message

Deletion successful.

2.2.7 Firmware

- [Get-IgelFirmware](#)(see page 61)

Get-IgelFirmware

Summary

Gets information on the firmwares registered with UMS.

Parameters

- **-Credentials**
Credential data, usually passed via a variable which it has been saved to at login
- **-firmwareID**
Numeric ID of the firmware. If omitted, information is retrieved on all firmwares.

Example Command Line

```
Get-IgelFirmware -Credentials $l -firmwareID 3
```

Example Output

```
id product version firmwareType
```

```
-- -----
```



3 IGEL Universal Desktop LX 5.09.100.01 LX

2.2.8 Server Status

- [Get-IgelRMServerStatus](#)(see page 62)

Get-IgelRMServerStatus

Summary

Gets information on the UMS Server and IMI. No authentication is necessary for this.

Parameters

- **-Servername**
Hostname of the UMS server with IMI
- **-IgnoreUntrustedCertificates**
If this is set to `$true` the Cmdlets will ignore errors in TLS/SSL certificate validation.

Example Command Line

```
Get-IgelRMServerStatus -Servername 172.30.91.158 -IgnoreUntrustedCertificates $true
```

Example Output

```
buildNumber : 26712
rmGuiServerVersion : 5.02.100.rc8
activeMQVersion : 5.6.0
derbyVersion : 10.12.1.1
serverUUID : a400f9da-333d-49d8-bb8d-2238543a4643
server : mhuber:8443
```



3 IMI API V3 Reference

- [What's New in IMI V3?](#)(see page 63)
- [Prerequisites](#)(see page 63)
- [Client Applications and Libraries](#)(see page 63)
- [Authentication](#)(see page 65)
- [Basic Data Types](#)(see page 67)
- [Resources](#)(see page 77)
- [Error Codes](#)(see page 189)

3.1 What's New in IMI V3?

- Asset Inventory Tracker (AIT) Resources:
 - [GET /v3/assetinfo](#)(see page 77)
 - [GET /v3/assethistory](#)(see page 85)
- A command for resetting thin clients to factory defaults:
[POST /v3/thinclients/?command=tcreset2facdefs](#)(see page 117)

3.2 Prerequisites

- IGEL OS *version 10.03.100* or newer
- UMS *version 5.07.100* or newer
- Networking: In order to use *IGEL Management Interface* you need to be able to reach the API host via the network and connect to its API port, TCP 8443 by default.
The base URL is
`https://[server]:8443/umsapi/`
The API is versioned, so e.g. the resources for version 3 are available at
`https://[server]:8443/umsapi/v3/`

If you are using the .NET framework for your client application, make sure to deploy version 4.5 or newer, as this includes the TLSv1.2 support required by IMI.

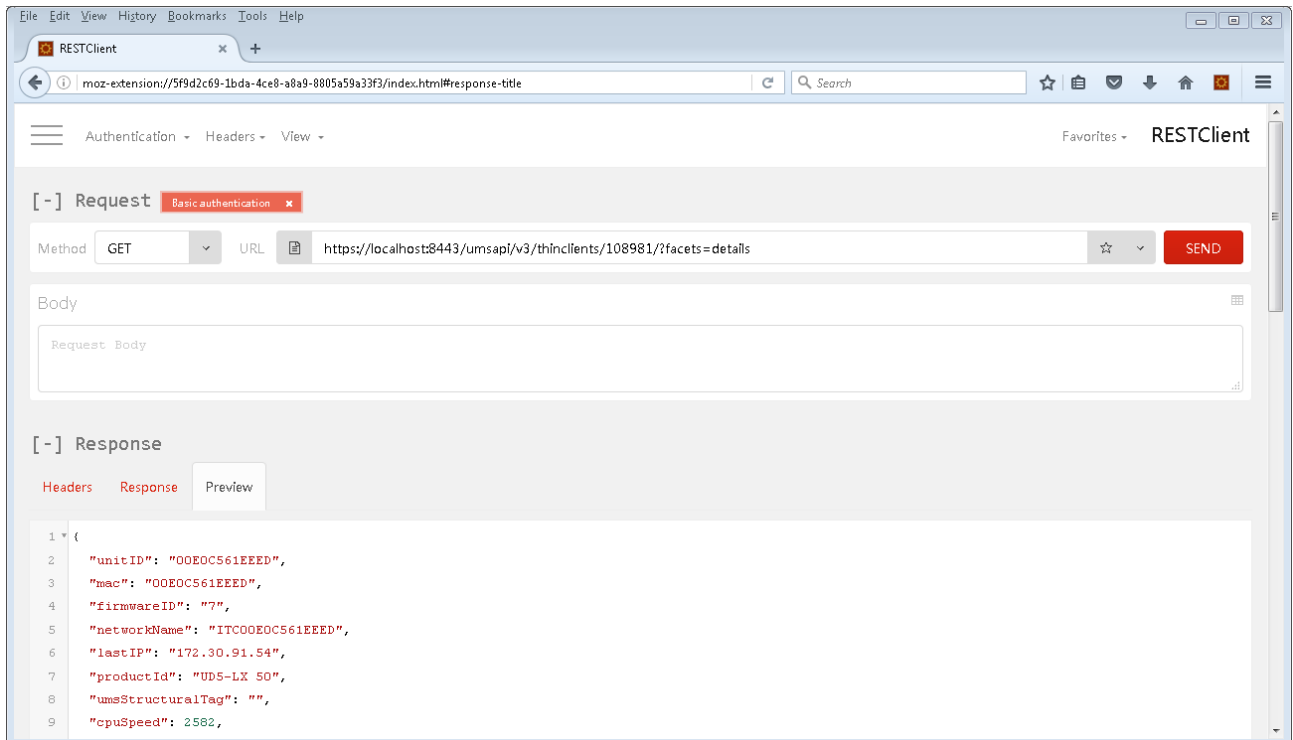
3.3 Client Applications and Libraries

3.3.1 Clients

The easiest way to try out the IGEL Management Interface is either

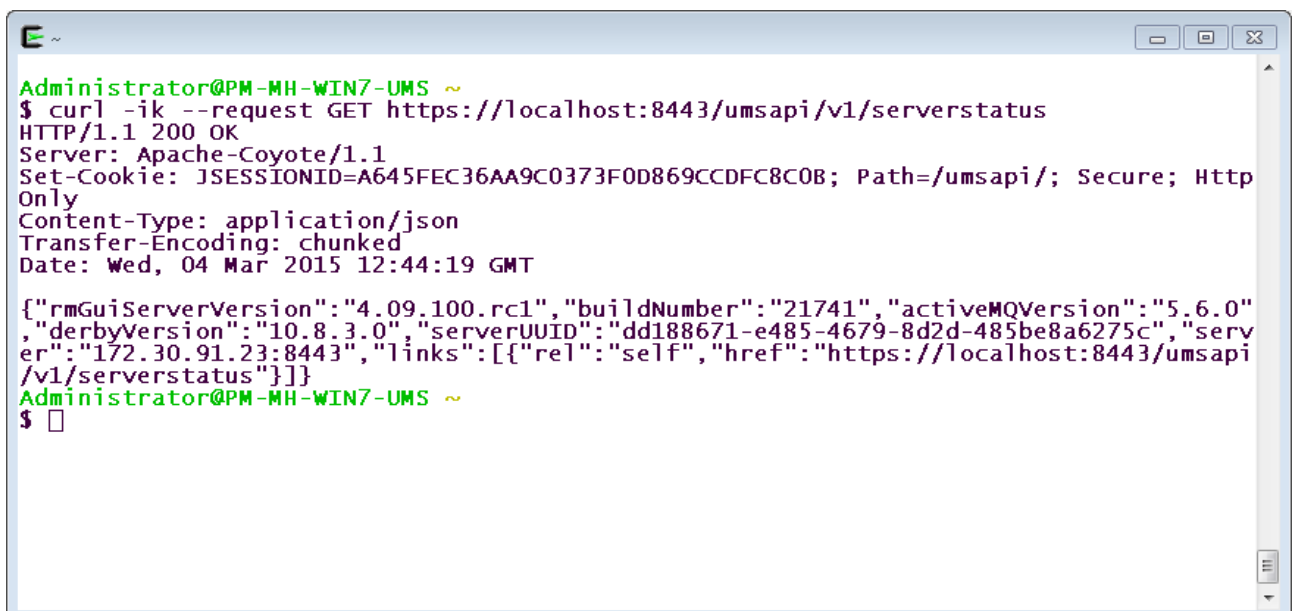
- with [RESTClient](#)¹¹, an add-on for Mozilla Firefox:

¹¹ <http://restclient.net/>



or

- with [cURL](http://curl.haxx.se/)¹², a commandline network client:



Both are licensed as Open Source software and are available free of charge.

¹² <http://curl.haxx.se/>



This guide uses *cURL* for examples, as the commandline makes all parameters visible in plain text. This need not stop you from using *RESTClient* if you are more comfortable with it. You can easily translate the commandline parameters into the fields of the *RESTClient* GUI.

3.3.2 Programming

Most programming languages provide an HTTP and an SSL/TLS module, either in their standard library or as an extension, and a *JSON* library (for the API data format) as well.

3.4 Authentication

You must be authenticated in order to use *IGEL Management Interface*, otherwise the server will return the HTTP status 401 "Unauthorized". Only querying the server status is allowed without authentication.

The login mechanism uses HTTP Basic Authentication (RFC 2617).

-
- [POST /v3/login](#)(see page 65)
 - [POST /v3/logout](#)(see page 66)

3.4.1 POST /v3/login

Summary

Authenticates the client in regard to the *IMI* server.

Resource URL

/v3/login

Example Request

```
curl \
--request POST \
--user 'admin:W00t' \
https://[server]:8443/umsapi/v3/login
```

Response Body

name	type	description
message	String	String containing the JSESSIONID



Example Response

200 OK

Set-Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC; Path=/umsapi/; Secure; HttpOnly

```
{"message": "JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC"}
```

You can maintain the session by sending the JSESSIONID in the Cookie header with every subsequent request. Some clients will do this automatically for you, e.g. the *RESTClient Firefox* add-on.

Cookie: JSESSIONID=3FB2F3F6A089FE9029DFD6DAFEF146DC

3.4.2 POST /v3/logout

Summary

Logs the client out from the *IMI* session.

Resource URL

/v3/logout

Example Request

```
curl \
--request POST \
--header 'Cookie: JSESSIONID=11048CDA77DE2B45BE1562C8EED67858' \
https://[server]:8443/umsapi/v3/logout
```

Example Response

200 OK

```
{"message": "Logout successful"}
```



3.5 Basic Data Types

- [ApiObject](#)(see page 67)
- [AssetInfoResource](#)(see page 68)
- [AssetDetailResource](#)(see page 69)
- [AssetCategory](#)(see page 70)
- [AssetHistoryResource](#)(see page 70)
- [TCResourceV3](#)(see page 71)
- [Assignment](#)(see page 74)
- [DirectoryResource](#)(see page 76)

3.5.1 ApiObject

Summary

This object is the most basic representation of thin clients, profiles, and directories. It is used in [Assignments](#)(see page 74).

Properties

Name	Type	Mandatory	Description
id	String	yes	The object's ID in the UMS database
type	String	yes	One of: <ul style="list-style-type: none">• tc• tcdirectory• profile• profiledirectory• masterprofile• masterprofiledirectory

Example

```
{  
  "id": "68257",  
  "type": "profile"  
}
```



3.5.2 AssetInfoResource

Summary

This object describes an asset. It references the thin client, it is connected to, one or more categories the asset belongs to, and one or more details about the asset.

Properties

Name	Type	Description
tcId	Integer	Internal ID of the thin client the asset is connected to. The value -1 indicates that the asset is currently not connected to a thin client.
assetId	Integer	Internal asset ID
uniqueId	String	(not used)
connector	String	(not used)
assetName	String	Name
assetVendor	String	Vendor
deviceId	String	USB device ID
categoryList	Array of AssetCategory (see page 70)	Categories the asset belongs to
detailList	Array of AssetDetailResource (see page 69)	Asset details

Example

```
{  
  "tcId": 47922,  
  "assetId": 49070,
```



```
"assetName": "M105 Optical Mouse",
"assetVendor": "Logitech, Inc.",
"deviceId": "c077",
"categoryList": {
"categoryList": [...]
},
"detailList": {
"detailList": [...]
}
}
```

3.5.3 AssetDetailResource

Summary

This object describes one detail of the asset.

Properties

Name	Type	Description
attrId	Integer	Attribute ID
detailKey	String	Key for the detail
detailValue	String	Value (and unit)
detailType	String	Data type of the detail

Example

```
{
"attrId": 49067,
"detailKey": "maxPower",
"detailValue": "100mA",
"detailType": "string"
}
```



3.5.4 AssetCategory

Summary

This object describes one category the asset belongs to.

Properties

Name	Type	Description
categoryId	Integer	Internal ID
categoryName	String	Category name

Example

```
{  
  "categoryId": 49068,  
  "categoryName": "Keyboard"  
}
```

3.5.5 AssetHistoryResource

Summary

This object describes an item (event) in the asset history.

Properties

Name	Type	Description
historyId	Integer	ID of the history item
unitId	String	UnitID of the thin client
assetId	Integer	Asset ID
eventTimeStamp	Timestamp	Unix timestamp of the event



event		one of: <ul style="list-style-type: none"> • add • remove • add_on_boot • remove_on_boot
-------	--	--

Example

```
{
  "historyId": 26,
  "unitId": "000BCA050027",
  "assetId": 49070,
  "eventTimeStamp": 1500372856000,
  "event": {
    "identifier": "remove"
  }
}
```

3.5.6 TCResourceV3

Summary

This object describes a thin client with its properties.

Properties

Name	Type	Description
id	String	Object ID in the UMS database
unitID	String	Unique unit ID
name	String	Network name
parentID	String	Name of parent directory
movedToBin	Boolean	True if in Recycling Bin
objectType		tc



links	Array	Links that make IMI discoverable for an HTTP spider
mac	String (12)	MAC address
firmwareID	String	Firmware ID
lastIP	String	Last known IP
batteryLevel	Integer	Level of the battery (if any)

With facets=details

Name	Type	Description
networkName	String	
site	String	Site
department	String	Department
costCenter	String	Cost center
comment	String	Comment
assetID	String	Asset ID of the thin client
inserviceDate	String	Placed in service on
serialNumber	String	Serial Number
productId	String	Product ID
umsStructuralTag	String	Structure Tag
cpuSpeed	Integer	CPU speed (MHz)
cpuType	String	CPU type
deviceType	String	Device type
deviceSerialNumber	String	Device serial number



osType	String	Operating system type
flashSize	Integer	Flash size (MB)
memorySize	Integer	Flash size (MB)
networkSpeed	Integer	Network speed
duplexMode	String	Duplex mode
graphicsChipset0	String	First Graphics chipset
graphicsMemorySize0	Integer	First Graphics memory (MB)
graphicsChipset1	String	Second Graphics chipset
graphicsMemorySize1	Integer	Second Graphics memory (MB)
monitorVendor1	String	First Monitor vendor
monitorModel1	String	First Monitor model
monitorSerialnumber1	String	First Monitor serial number
monitorSize1	Number	First Monitor size
monitorNativeResolution1	String	First Monitor native resolution
monitor1YearOfManufacture	String	First Monitor year of manufacture
monitor1WeekOfManufacture	String	First Monitor week of manufacture
monitorVendor2	String	Second Monitor vendor
monitorModel2	String	Second Monitor model
monitorSerialnumber2	String	Second Monitor serial number
monitorSize2	Number	Second Monitor size
monitorNativeResolution2	String	Second Monitor native resolution



monitor2YearOfManufacture	String	Second Monitor year of manufacture
monitor2WeekOfManufacture	String	Second Monitor week of manufacture
biosVendor	String	BIOS vendor
biosVersion	String	BIOS version
biosDate	String	BIOS date
totalUsagetime	String	Total usage time
totalUptime	String	Total uptime
lastBoottime	String	Last boot time
lastContact	String	Date and time of the last contact between UMS and endpoint device (UMS 6.07 or higher)
configChanges		PENDING APPLIED (UMS 6.07 or higher)

With facets=online

online	Boolean	Online status
--------	---------	---------------

With facets=shadow

shadowSecret	String	Certificate and password for Secure VNC
--------------	--------	---

3.5.7 Assignment

Summary

An Assignment links [ApiObjects](#)(see page 67), e.g. profiles to thin clients, or profiles to thin client directories.



Properties

Name	Type	Mandatory	Description
assignee	ApiObject (see page 67)	yes	The ApiObject (see page 67) that is assigned
receiver	ApiObject (see page 67)	yes	The ApiObject (see page 67) that is assigned to
assignmentPosition	Integer	no	Position for the order in which profiles are applied
links	Array	no	Links that make IMI discoverable for an HTTP spider

Example

```
[
  {
    "assignee": {
      "id": "68257",
      "type": "profile"
    },
    "receiver": {
      "id": "48335",
      "type": "tc"
    },
    "assignmentPosition": 0,
    "links": [
      {
        "rel": "assigned",
        "href": "https://172.30.91.227:8443/umsapi/v2/profiles/68257"
      },
      {
        "rel": "receiver",
```



```

    "href": "https://172.30.91.227:8443/umsapi/v2/thinclients/48335"
  },
  {
    "rel": "self",
    "href": "https://172.30.91.227:8443/umsapi/v2/profiles/68257/assignments/thinclients/48335"
  }
]
}
]

```

3.5.8 DirectoryResource

Properties

Name	Type	Mandatory	Description
id	String	yes	Object ID in the UMS database
name	String	yes	Directory name
parentID	String	no	ID of parent directory
movedToBin	Boolean	no	True if in Recycling Bin
objectType		no	One of: <ul style="list-style-type: none"> • tcdirectory • profiledirectory • masterprofiledirectory
links	Array	no	Links that make IMI discoverable for an HTTP spider

With facets=children

DirectoryChildren	Array		Array of objects (thin clients, directories or profiles) contained in this directory
-------------------	-------	--	--



3.6 Resources

- [Asset Information](#)(see page 77)
- [Thin Client](#)(see page 91)
- [Profile](#)(see page 118)
- [Master Profile](#)(see page 131)
- [Thin Client Directory](#)(see page 144)
- [Profile Directory](#)(see page 160)
- [Master Profile Directory](#)(see page 174)
- [Firmware Information](#)(see page 185)
- [Server Information](#)(see page 188)

3.6.1 Asset Information

- [GET /v3/assetinfo](#)(see page 77)
- [GET /v3/assetinfo/assets/{assetId}](#)(see page 81)
- [GET /v3/assetinfo/thinclients/{tclid}](#)(see page 83)
- [GET /v3/assethistory](#)(see page 85)
- [GET /v3/assethistory/assets/{assetId}](#)(see page 88)
- [GET /v3/assethistory/thinclients/{unitId}](#)(see page 89)

GET /v3/assetinfo

Summary

Gets information on all assets known to the UMS instance.

Resource URL

/v3/assetinfo

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/assetinfo
```

Response Type

Returns an array of [AssetInfoResource](#)(see page 68).

Example Response

```
{
  "assetinfos": [
```



```
{
  "tcId": 47922,
  "assetId": 49067,
  "assetName": "Elite Keyboard",
  "assetVendor": "Hewlett-Packard",
  "deviceId": "034a",
  "categoryList": {
    "categoryList": [
      {
        "categoryId": 49068,
        "categoryName": "Keyboard"
      },
      {
        "categoryId": 49069,
        "categoryName": "Human Interface Device"
      }
    ]
  },
  "detailList": {
    "detailList": [
      {
        "attrId": 49067,
        "detailKey": "maxPower",
        "detailValue": "100mA",
        "detailType": "string"
      },
      {
        "attrId": 49067,
        "detailKey": "speed",
        "detailValue": "1.5",
        "detailType": "string"
      },
      {
        "attrId": 49067,
```



```
"detailKey": "custom_productName",
"detailValue": "HP_Elite_USB_Keyboard",
"detailType": "string"
},
{
  "attrId": 49067,
  "detailKey": "custom_vendorName",
  "detailValue": "Chicony",
  "detailType": "string"
},
{
  "attrId": 49067,
  "detailKey": "revision",
  "detailValue": "0121",
  "detailType": "string"
}
]
}
},
{
  "tcId": 47922,
  "assetId": 49070,
  "assetName": "M105 Optical Mouse",
  "assetVendor": "Logitech, Inc.",
  "deviceId": "c077",
  "categoryList": {
    "categoryList": [
      {
        "categoryId": 49071,
        "categoryName": "Mouse"
      }
    ]
  },
  "detailList": {
```



```
"detailList": [  
  {  
    "attrId": 49070,  
    "detailKey": "maxPower",  
    "detailValue": "100mA",  
    "detailType": "string"  
  },  
  {  
    "attrId": 49070,  
    "detailKey": "speed",  
    "detailValue": "1.5",  
    "detailType": "string"  
  },  
  {  
    "attrId": 49070,  
    "detailKey": "custom_productName",  
    "detailValue": "USB_Optical_Mouse",  
    "detailType": "string"  
  },  
  {  
    "attrId": 49070,  
    "detailKey": "custom_vendorName",  
    "detailValue": "Logitech",  
    "detailType": "string"  
  },  
  {  
    "attrId": 49070,  
    "detailKey": "revision",  
    "detailValue": "7200",  
    "detailType": "string"  
  }  
]  
}
```




```
],  
  "links": []  
}
```

GET /v3/assetinfo/assets/{assetId}

Summary

Gets information on a specific asset.

Resource URL

/v3/assetinfo/assets/{assetId}

Example Request

```
curl \  
--request GET \  
https://[server]:8443/umsapi/v3/assetinfo/assets/49070
```

Request Path Variables

Name	Type	Mandatory	Description
assetId	String	yes	asset ID

Response Type

Returns an array of [AssetInfoResource](#)(see page 68).

Example Response

```
{  
  "assetinfos": [  
    {  
      "tcId": 47922,  
      "assetId": 49070,  
      "assetName": "M105 Optical Mouse",  
      "assetVendor": "Logitech, Inc.",  
      "deviceId": "c077",  
      "categoryList": {  
        "categoryList": [  

```



```
{
  "categoryId": 49071,
  "categoryName": "Mouse"
}
],
{
  "detailList": {
    "detailList": [
      {
        "attrId": 49070,
        "detailKey": "maxPower",
        "detailValue": "100mA",
        "detailType": "string"
      },
      {
        "attrId": 49070,
        "detailKey": "speed",
        "detailValue": "1.5",
        "detailType": "string"
      },
      {
        "attrId": 49070,
        "detailKey": "custom_productName",
        "detailValue": "USB_Optical_Mouse",
        "detailType": "string"
      },
      {
        "attrId": 49070,
        "detailKey": "custom_vendorName",
        "detailValue": "Logitech",
        "detailType": "string"
      },
      {
        "attrId": 49070,
```



```
"detailKey": "revision",
"detailValue": "7200",
"detailType": "string"
}
]
}
],
"links": []
}
```

GET /v3/assetinfo/thinclients/{tcId}

Summary

Gets asset information on a specific thin client.

Resource URL

/v3/assetinfo/thinclients/{tcId}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/assetinfo/thinclients/47922
```

Request Path Variables

Name	Type	Mandatory	Description
tcId	String	yes	thin client ID

Response Type

Returns an array of [AssetInfoResource](#)(see page 68).

Example Response

```
{
  "assetinfos": [
    {
```



```
"tcId": 47922,
"assetId": 49072,
"assetName": "Bluetooth Dongle (HCI mode)",
"assetVendor": "Cambridge Silicon Radio, Ltd",
"deviceId": "0001",
"categoryList": {
  "categoryList": [
    {
      "categoryId": 49073,
      "categoryName": "Bluetooth"
    }
  ]
},
"detaillList": {
  "detaillList": [...]
},
{
  "tcId": 47922,
  "assetId": 49070,
  "assetName": "M105 Optical Mouse",
  "assetVendor": "Logitech, Inc.",
  "deviceId": "c077",
  "categoryList": {
    "categoryList": [
      {
        "categoryId": 49071,
        "categoryName": "Mouse"
      }
    ]
  },
  "detaillList": {
    "detaillList": [...]
  }
}
```



```
},
{
  "tcId": 47922,
  "assetId": 49067,
  "assetName": "Elite Keyboard",
  "assetVendor": "Hewlett-Packard",
  "deviceId": "034a",
  "categoryList": {
    "categoryList": [
      {
        "categoryId": 49068,
        "categoryName": "Keyboard"
      },
      {
        "categoryId": 49069,
        "categoryName": "Human Interface Device"
      }
    ]
  },
  "detailList": {
    "detailList": [...]
  }
],
"links": []
}
```

GET /v3/assethistory

Summary

Gets the complete asset history of the UMS instance.

Resource URL

/v3/assethistory



Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/assethistory
```

Response Type

Returns an array of [AssetHistoryResource](#)(see page 70).

Example Response

```
{
  "assethistories": [
    {
      "historyId": 2,
      "unitId": "000BCA050027",
      "assetId": 49067,
      "eventTimeStamp": 1500012051000,
      "event": {
        "identifier": "add_on_boot"
      }
    },
    {
      "historyId": 3,
      "unitId": "000BCA050027",
      "assetId": 49070,
      "eventTimeStamp": 1500012051000,
      "event": {
        "identifier": "add_on_boot"
      }
    },
    {
      "historyId": 4,
      "unitId": "000BCA050027",
      "assetId": 49072,
      "eventTimeStamp": 1500012123000,
```



```
"event": {
  "identifier": "add"
},
{
  "historyId": 5,
  "unitId": "000BCA050027",
  "assetId": 49072,
  "eventTimeStamp": 1500012735000,
  "event": {
    "identifier": "remove"
  },
  {
    "historyId": 6,
    "unitId": "000BCA050027",
    "assetId": 49083,
    "eventTimeStamp": 1500012774000,
    "event": {
      "identifier": "add"
    },
    {
      "historyId": 7,
      "unitId": "000BCA050027",
      "assetId": 49072,
      "eventTimeStamp": 1500274346000,
      "event": {
        "identifier": "add"
      }
    },
    "links": []
  }
```



GET /v3/assethistory/assets/{assetId}

Summary

Gets the asset history for a specific asset.

Resource URL

/v3/assethistory/assets/{assetId}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/assethistory/assets/49070
```

Request Path Variables

Name	Type	Mandatory	Description
assetId	String	yes	asset ID

Response Type

Returns an array of [AssetHistoryResource](#)(see page 70).

Example Response

```
{
  "assethistories": [
    {
      "historyId": 3,
      "unitId": "000BCA050027",
      "assetId": 49070,
      "eventTimeStamp": 1500012051000,
      "event": {
        "identifier": "add_on_boot"
      }
    },
    {
      "historyId": 26,
```




```
"unitId": "000BCA050027",
"assetId": 49070,
"eventTimeStamp": 1500372856000,
"event": {
  "identifier": "remove"
},
{
  "historyId": 29,
  "unitId": "000BCA050027",
  "assetId": 49070,
  "eventTimeStamp": 1500380032000,
  "event": {
    "identifier": "add"
  }
},
"links": []
}
```

GET /v3/assethistory/thinclients/{unitId}

Summary

Gets the asset history for a specific thin client.

Resource URL

/v3/assethistory/thinclients/{unitId}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/assethistory/thinclients/000BCA050027
```

Request Path Variables

Name	Type	Mandatory	Description
------	------	-----------	-------------



unitId	String	yes	unit ID of the endpoint device
--------	--------	-----	--------------------------------

Response Type

Returns an array of [AssetHistoryResource](#)(see page 70).

Example Response

```
{
  "assethistories": [
    {
      "historyId": 2,
      "unitId": "000BCA050027",
      "assetId": 49067,
      "eventTimeStamp": 1500012051000,
      "event": {
        "identifier": "add_on_boot"
      }
    },
    {
      "historyId": 3,
      "unitId": "000BCA050027",
      "assetId": 49070,
      "eventTimeStamp": 1500012051000,
      "event": {
        "identifier": "add_on_boot"
      }
    },
    {
      "historyId": 4,
      "unitId": "000BCA050027",
      "assetId": 49072,
      "eventTimeStamp": 1500012123000,
      "event": {
        "identifier": "add"
      }
    }
  ]
}
```



```

},
{
  "historyId": 5,
  "unitId": "000BCA050027",
  "assetId": 49072,
  "eventTimeStamp": 1500012735000,
  "event": {
    "identifier": "remove"
  }
}
],
"links": []
}

```

3.6.2 Thin Client

- [GET /v3/thinclients](#)(see page 91)
- [GET /v3/thinclients?facets=details](#)(see page 93)
- [GET /v3/thinclients?facets=online](#)(see page 96)
- [GET /v3/thinclients?facets=shadow](#)(see page 98)
- [GET /v3/thinclients/{tcId}](#)(see page 100)
- [GET /v3/thinclients/{tcId}?facets=details](#)(see page 101)
- [GET /v3/thinclients/{tcId}?facets=online](#)(see page 103)
- [GET /v3/thinclients/{tcId}?facets=shadow](#)(see page 104)
- [GET /v3/thinclients/{tcId}/assignments/profiles](#)(see page 106)
- [PUT /v3/thinclients/{tcId}/assignments/profiles](#)(see page 109)
- [PUT /v3/thinclients](#)(see page 110)
- [PUT /v3/thinclients/{tcId}](#)(see page 111)
- [DELETE /v3/thinclients/{tcId}](#)(see page 113)
- [DELETE /v3/thinclients/{tcId}/deletetcoffline](#)(see page 114)
- [POST /v3/thinclients/?command={reboot|shutdown|wakeup}](#)(see page 115)
- [POST /v3/thinclients?command=settings2tc](#)(see page 116)
- [POST /v3/thinclients/?command=tcreset2facdefs](#)(see page 117)

GET /v3/thinclients

Summary

Gets information on all thin clients registered with the UMS instance.



This method will also list thin clients that are located in the **Recycle Bin** ("movedToBin": "true"), but you will not be able to call methods on those clients.

Resource URL

/v3/thinclients

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/thinclients
```

Response Type

Returns an Array of [TCResourceV3](#)(see page 71).

Example Response

```
[
  {
    "unitID": "00E0C514325B",
    "mac": "00E0C514325B",
    "firmwareID": "6",
    "lastIP": "172.30.92.26",
    "batteryLevel": 0,
    "id": "95",
    "name": "ITC00E0C514325B",
    "parentID": "-1",
    "movedToBin": false,
    "objectType": "tc",
    "links": []
  },
  {
    "unitID": "000BCA050027",
    "mac": "000BCA050027",
    "firmwareID": "6",
    "lastIP": "172.30.91.87",
    "batteryLevel": 0,
```



```
"id": "47922",
"name": "ITC000BCA050027",
"parentID": "-1",
"movedToBin": false,
"objectType": "tc",
"links": []
}
]
```

GET /v3/thinclients?facets=details

Summary

Gets detailed information on all thin clients.

This method will also list thin clients that are located in the **Recycle Bin** ("movedToBin": "true"), but you will not be able to call methods on those clients.

Resource URL

/v3/thinclients?facets=details

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/thinclients?facets=details
```

Response Type

Returns an Array of [TCResourceV3](#) (see page 71) with full details.

Example Response

```
[
{
  "unitID": "00E0C514325B",
  "mac": "00E0C514325B",
  "firmwareID": "6",
  "networkName": "ITC00E0C514325B",
  "lastIP": "172.30.92.26",
```



```
"productId": "UD3-LX 50cps",
"umsStructuralTag": "",
"cpuSpeed": 1200,
"cpuType": "AMD GX-412HC SOC with Radeon(TM) R3E Graphics",
"deviceType": "IGEL M340C",
"deviceSerialNumber": "14D3E3C02B16160323M",
"osType": "IGEL Linux 10 (Kernel Version 4.10.17)",
"flashSize": 1875,
"memorySize": 1705,
"networkSpeed": 1000,
"graphicsChipset0": "ATI MULLINS",
"graphicsChipset1": "",
"monitorVendor1": "Samsung Electric Company",
"monitorModel1": "SyncMaster",
"monitorSerialnumber1": "HMJBC00558",
"monitorSize1": 24,
"monitorNativeResolution1": "1920 x 1200",
"monitor1YearOfManufacture": "2011",
"monitor1WeekOfManufacture": "51",
"monitorVendor2": "",
"monitorModel2": "",
"monitorSerialnumber2": "",
"monitorSize2": 0,
"monitorNativeResolution2": "",
"monitor2YearOfManufacture": "",
"monitor2WeekOfManufacture": "",
"biosVendor": "Insyde Corp.",
"biosVersion": "M340C V:3.3.13-12162015",
"biosDate": "12/16/2015",
"totalUsagetime": "20",
"totalUptime": "1500380741",
"lastBoottime": "2017-07-18 14:25",
"batteryLevel": -1,
"id": "95",
```



```

"name": "ITC00E0C514325B",
"parentID": "-1",
"movedToBin": false,
"objectType": "tc",
"links": [],
},
{
"unitID": "000BCA050027",
"mac": "000BCA050027",
"firmwareID": "6",
"networkName": "ITC000BCA050027",
"lastIP": "172.30.91.87",
"productId": "UD2-LX 40acps",
"umsStructuralTag": "",
"cpuSpeed": 1466,
"cpuType": "Intel(R) Atom(TM) CPU E3815 @ 1.46GHz",
"deviceType": "IGEL D220",
"deviceSerialNumber": "1502GVAAA2965",
"osType": "IGEL Linux 10 (Kernel Version 4.10.17)",
"flashSize": 1875,
"memorySize": 1894,
"networkSpeed": 1000,
"graphicsChipset0": "INTEL HD Graphics (Baytrail)",
"graphicsChipset1": "",
"monitorVendor1": "Samsung Electric Company",
"monitorModel1": "SyncMaster",
"monitorSerialnumber1": "HMJBC00558",
"monitorSize1": 24,
"monitorNativeResolution1": "1920 x 1200",
"monitor1YearOfManufacture": "2011",
"monitor1WeekOfManufacture": "51",
"monitorVendor2": "",
"monitorModel2": "",
"monitorSerialnumber2": "",

```



```

"monitorSize2": 0,
"monitorNativeResolution2": "",
"monitor2YearOfManufacture": "",
"monitor2WeekOfManufacture": "",
"biosVendor": "INSYDE Corp.",
"biosVersion": "D220-002",
"biosDate": "02/01/2016",
"totalUsagetime": "1739549000",
"totalUptime": "8856000",
"lastBoottime": "2017-07-18 11:55",
"batteryLevel": -1,
"id": "47922",
"name": "ITC000BCA050027",
"parentID": "-1",
"movedToBin": false,
"objectType": "tc",
"links": []
}
]

```

GET /v3/thinclients?facets=online

Summary

Gets the online status for all thin clients.

This method will also list thin clients that are located in the **Recycle Bin** ("movedToBin": "true"), but you will not be able to call methods on those clients.

When you send this request, UMS cannot answer it from data in the database alone, but makes a network connection to each thin client, which may take some time.

Resource URL

/v3/thinclients?facets=online

Example Request

```
curl \
```




```
--request GET \  
https://[server]:8443/umsapi/v3/thinclients?facets=online
```

Response Type

Returns an Array of [TCResourceV3](#)(see page 71) with their online status.

Example Response

```
[  
  {  
    "unitID": "00E0C514325B",  
    "mac": "00E0C514325B",  
    "firmwareID": "6",  
    "lastIP": "172.30.92.26",  
    "online": false,  
    "batteryLevel": 0,  
    "id": "95",  
    "name": "ITC00E0C514325B",  
    "parentID": "-1",  
    "movedToBin": false,  
    "objectType": "tc",  
    "links": []  
  },  
  {  
    "unitID": "000BCA050027",  
    "mac": "000BCA050027",  
    "firmwareID": "6",  
    "lastIP": "172.30.91.87",  
    "online": true,  
    "batteryLevel": 0,  
    "id": "47922",  
    "name": "ITC000BCA050027",  
    "parentID": "-1",  
    "movedToBin": false,  
    "objectType": "tc",  
    "links": []  
  }  
]
```



```
}
]
```

GET /v3/thinclients?facets=shadow

Summary

Gets the certificate and password for Secure VNC for all thin clients.

This method will also list thin clients that are located in the **Recycle Bin** ("movedToBin": "true"), but you will not be able to call methods on those clients.

Learn more about shadowing from the How-To document "[Using Secure VNC via IGEL Management Interface \(IMI\)](#)"¹³.

Resource URL

/v3/thinclients?facets=shadow

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/thinclients?facets=shadow
```

Response Type

Returns an Array of [TCResourceV3](#) (see page 71) with the shadowSecret object, consisting of the certificate and password fields.

Example Response

```
[
{
  "unitID": "00E0C514325B",
  "mac": "00E0C514325B",
  "firmwareID": "6",
  "lastIP": "172.30.92.26",
  "batteryLevel": 0,
  "shadowSecret": {
    "certificate": "-----BEGIN CERTIFICATE-----
```

¹³ <http://edocs.igel.com/10204968.htm>



MIIE1jCCAr6gAwIBAgIEBC0a4zANBgkqhkiG9w0BAQ0FADAtMQswCQYDVQQGEwJERTEPMA0GA1UEBxMGQnJlbWVudMQ0wCwYDVQQKEwRJR0VMMB4XDTE0MDEwMTAwMDAwMVoxDTM3MDEwMTAwMDAwMVowLTELMAkGA1UEBhMCREUxDzANBgNVBACTBkJyZW1lbjENMAsGA1UEChMESUdFTDCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgocGgIBAJw6ki ftBFhAnFXGfzMnuhiSYm9QrGFGvryvNpPzEK8f8Svcp62R+jv3bhtlFJcWQGf4xn4xRoQUzj48m5Y1Sg12oY/YBNt5xLxKLTeq0HGurpYVan40zoT0FRG+ygpXWUh0MvPfFYscokWo/7Id5EycfBefVwvx6Z09z0b112J8qgmo5/aenWJph70dJbKWsh2ojXzjU6tkrMvV80V+68TN0c5I8V2wIzlgEqNnpKCZLkSbRjzhd6FMoY0RdqqCMLuMALuDoKuj5Z29tGNX1YYnIt7w+DmB18g8hYwBLztLhMqX6pNBfEgi3VLqjUOSiIFmfzGT9ZpuCNMfvpXzCWKhwc8WPWzLPNFF9cMI+rYyPWG0mhSuhIJsHvRbddfm6nte8Wt7cc2Q0Ha89b1Mwmy4/LKT810Ry0Ge8t66eDrxkGv2dlmimUJ2Tn6E92b9WyStLC+FDqEecPC46P8UCEf4c9Ed+MiwrQzwIFBlqAMqljtG1StmwEoYJ8WLazl5wG9QpS0zy/4JKTqjm5pLfUYA2xIfnP1h/9ESD7AYU47PNZF/KRh20VCMBHeV79V7en/w0NJPeack9ug2XNFWHSARS2Bgb3KAXdtoicBIcjZAtNpm1W5N0aZ70l4zngV0V45X0GwXin+7oYXoLaMfi7pm0EnZmBNWmPSBdjRIQChAgMA/zkwDQYJKoZIhvcNAQENBQADggIBAImNVT+x4iprrlBdqChLCK1dhiozzbYBkRQGE8HXryIzCPY+t0z0N0Zr1xAiL8Iczy1DVZRHsAw5CNSWnSnfKHdd/tDV9aCpITfWUq0b22hrG9wK9P5XjZhBusc9BBX1oNL/CiD+Hschhs0qpdl4MMWpH6aSdoK+qxTW8+8300CiLHHZQuB8UkP/XSjr/70CmOuqd71QrfdbqrqTzdRmzhKg6SnwUR9j3nToym0iwaZM8ucR/gbN39mz0RiqiGydvjJ/LJgpV5viERLCK1iUua2SJ4+5NRByS//JfDt90U8YoGXAohMjwz55Arc8CJvd2M0W0v00eyIMj7EJzhTDcxS5mmLrl6BrsFC37qgFiSsjffo0XnsZMkMC/edUyGSqpJ4N5/huyYkoP/C70SQTXgsDE8gN4kESIWkzHLJCnP/A2A4jaHlmG8onZtR92xLSEksGGyAAp19Rx9510Rqo27+qooJTQ0D/0vjyf9i+dn9x3cczpmDsfn9ucu5Bd9TCNlP84pIw7rEyhA+otly47dgC6uL7digv0lfUtwenOgnNWYON9Y201SLiF5lyFpEmxq9m/bo6Vci2EWgf4D6Ib+R6ZpRUFvBZUHp6rfXl9Qi1bD5qM4ItJavl7XGk7CyGyF0/SCv3Sx0KP1d54poRY/iXeJ5LES0LpAurXS8/yX4i

-----END CERTIFICATE-----",

"password": "1e629a89-b1bb-471a-9318-5cf966267337"

},

"id": "95",

"name": "ITC00E0C514325B",

"parentID": "-1",

"movedToBin": false,

"objectType": "tc",



```

"links": []
},
{
  "unitID": "000BCA050027",
  "mac": "000BCA050027",
  "firmwareID": "6",
  "lastIP": "172.30.91.87",
  "batteryLevel": 0,
  "shadowSecret": {
    "certificate": "-----BEGIN CERTIFICATE-----
MIIE2jCCAsKgAwIBAgIIVK2HBBbVnSwwDQYJKoZIhvcNAQENBQAwLElMAkGA1U
[...]
v7vVw==
-----END CERTIFICATE-----",
    "password": "0fc16140-fe51-4f3a-9b33-e0aa4b21ce69"
  },
  "id": "47922",
  "name": "ITC000BCA050027",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tc",
  "links": []
}
]

```

GET /v3/thinclients/{tcId}

Summary

Gets information on the specified thin client.

Resource URL

/v3/thinclients/{tcId}

Example Request

```
curl \
```



```
--request GET \  
https://[server]:8443/umsapi/v3/thinclients/95
```

Request Path Variables

Name	Type	Mandatory	Description
tcId	String	yes	thin client ID

Response Type

Returns a [TCResourceV3](#)(see page 71).

Example Response

```
{  
  "unitID": "00E0C514325B",  
  "mac": "00E0C514325B",  
  "firmwareID": "6",  
  "lastIP": "172.30.92.26",  
  "batteryLevel": 0,  
  "id": "95",  
  "name": "ITC00E0C514325B",  
  "parentID": "-1",  
  "movedToBin": false,  
  "objectType": "tc",  
  "links": []  
}
```

GET /v3/thinclients/{tcId}?facets=details

Summary

Gets detailed information on the specified thin client.

Resource URL

/v3/thinclients/{tcId}?facets=details

Example Request

```
curl \  

```



```
--request GET \
https://[server]:8443/umsapi/v3/thinclients/95?facets=details
```

Request Path Variables

Name	Type	Mandatory	Description
tcId	String	yes	thin client ID

Response Type

Returns a [TCResourceV3](#)(see page 71) with full details.

Example Response

```
{
  "unitID": "00E0C514325B",
  "mac": "00E0C514325B",
  "firmwareID": "6",
  "networkName": "ITC00E0C514325B",
  "lastIP": "172.30.92.26",
  "productId": "UD3-LX 50cps",
  "umsStructuralTag": "",
  "cpuSpeed": 1200,
  "cpuType": "AMD GX-412HC SOC with Radeon(TM) R3E Graphics",
  "deviceType": "IGEL M340C",
  "deviceSerialNumber": "14D3E3C02B16160323M",
  "osType": "IGEL Linux 10 (Kernel Version 4.10.17)",
  "flashSize": 1875,
  "memorySize": 1705,
  "networkSpeed": 1000,
  "graphicsChipset0": "ATI MULLINS",
  "graphicsChipset1": "",
  "monitorVendor1": "Samsung Electric Company",
  "monitorModel1": "SyncMaster",
  "monitorSerialnumber1": "HMJBC00558",
  "monitorSize1": 24,
  "monitorNativeResolution1": "1920 x 1200",
```



```
"monitor1YearOfManufacture": "2011",
"monitor1WeekOfManufacture": "51",
"monitorVendor2": "",
"monitorModel2": "",
"monitorSerialnumber2": "",
"monitorSize2": 0,
"monitorNativeResolution2": "",
"monitor2YearOfManufacture": "",
"monitor2WeekOfManufacture": "",
"biosVendor": "Insyde Corp.",
"biosVersion": "M340C V:3.3.13-12162015",
"biosDate": "12/16/2015",
"totalUsagetime": "20",
"totalUptime": "1500380741",
"lastBoottime": "2017-07-18 14:25",
"batteryLevel": -1,
"id": "95",
"name": "ITC00E0C514325B",
"parentID": "-1",
"movedToBin": false,
"objectType": "tc",
"links": []
}
```

GET /v3/thinclients/{tcId}?facets=online

Summary

Gets information on the specified thin client with its online status.

When you send this request, UMS cannot answer it from data in the database alone, but makes a network connection to the thin client, which may take some time.

Resource URL

/v3/thinclients/{tcId}?facets=online



Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/thinclients/95?facets=online
```

Request Path Variables

Name	Type	Mandatory	Description
tcId	String	yes	thin client ID

Response Type

Returns a [TCResourceV3](#)([see page 71](#)) with its online status.

Example Response

```
{
  "unitID": "00E0C514325B",
  "mac": "00E0C514325B",
  "firmwareID": "6",
  "lastIP": "172.30.92.26",
  "online": false,
  "batteryLevel": 0,
  "id": "95",
  "name": "ITC00E0C514325B",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tc",
  "links": []
}
```

GET /v3/thinclients/{tcId}?facets=shadow

Summary

Gets shadowing information on the specified thin client.

Learn more about shadowing from the How-To document "[Using Secure VNC via IGEL Management Interface \(IMI\)](#)"([see page 193](#)).



Resource URL

/v3/thinclients/{tcId}?facets=shadow

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/thinclients/95?facets=shadow
```

Request Path Variables

Name	Type	Mandatory	Description
tcId	String	yes	thin client ID

Response Type

Returns a [TCResourceV3](#)(see [page 71](#)) with the shadowSecret object, consisting of the certificate and password fields.

Example Response

```
{
  "unitID": "00E0C514325B",
  "mac": "00E0C514325B",
  "firmwareID": "6",
  "lastIP": "172.30.92.26",
  "batteryLevel": 0,
  "shadowSecret": {
    "certificate": "-----BEGIN CERTIFICATE-----
MIIE1jCCAr6gAwIBAgIEBC0a4zANBgkqhkiG9w0BAQ0FADAtMQswCQYDVQQGEwJE
RTEPMA0GA1UEBxMGQnJlbWVuMQ0wCwYDVQQKEwRJR0VMMA4XDTE0MDEwMTAwMDAw
MVoXDTM3MDEwMTAwMDAwMVowLTELMAkGA1UEBhMCREUxDzANBgNVBACTBkZyZW1l
bjENMA5GA1UEChMESUdFTDCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIB
AJw6ki ftBFhAnFXGfzMnuhiSYm9QrGFGvryvNpPzEK8f8Svcp62R+jv3bhtlFJcW
QGf4xn4xRoQUzj48m5Y1Sg12oY/YBNt5xLxKLTeq0HGurpYVan40zoT0FRG+ygpX
WUh0MvPFFYscokWo/7Id5EycfBefVwvx6Z09z0b112J8qgmo5/aenWJph70dJbKW
sH2ojXzjU6tkrMVv80V+68TN0c5I8V2wIzlgEqNnpKCZLkSbRjzhd6FMoY0RdqqC
MLuMALuDoKuj5Z29tGNX1YYnIt7w+DmB18g8hYwbLztLhMqX6pNBfEgi3VLqjUOS
```



```
iIFmfzGT9ZpuCNMfvpXzCWKhwc8WPWzLPNFF9cMI+rYyPWG0mhSuhIJshvRbddfM
6nte8Wt7cc2Q0Ha89b1Mwmy4/LKT810Ry0Ge8t66eDrxkGv2dLmimUJ2Tn6E92b9
WyStLC+FDqEecPC46P8UCEf4c9Ed+MiWRQzwIFBlqAMqljtG1StmwEoYJ8WLazL5
wG9QpS0zy/4JKTqjm5pLfUYA2xIfnP1h/9ESD7AYU47PNZF/KRh20VCMBHeV79V7
en/w0NJPeacK9ug2XNFWHSARS2Bgb3KAXdtoicBIcjZAtnpm1W5N0aZ70l4zngV0
V45X0Gwxin+7oYXoLaMfi7pm0EnZnmBNWmPSBdjRIQChAgMA/zkwDQYJKoZIhvcN
AQENBQADggIBAImNVT+x4iprrlBdqChLCK1dhiozzbYBkRQGE8HXryIzCPY+t0zO
N0Zr1xAIL8Iczy1DVZRHsAw5CNSWnSnfKHdd/tDV9aCpITfWUq0b22hrG9wK9P5X
jZhBusc9BBX1oNL/CiD+Hschhs0qpdL4MMWpH6aSdoK+qxTW8+8300CiLHHzQuB8
UkP/XSjr/70Cm0uqd71QrfdbRqTzdRmzhKg6SnwUR9j3nToym0iwaZM8ucR/gbN3
9mz0RiqiGydvjJ/LJgpV5viERLCK1iUua2SJ4+5NRByS//JfDt90U8YoGXAohMj
wz55Arc8CJvd2M0W0v00eyIMj7EJzhTDcxS5mmLrL6BrSFC37qgFiSsjffo0XnsZ
MkMC/edUyGSqpJ4N5/huyYkoP/C70SQTXgsDE8gN4kESIWkzHLJCnP/A2A4jaHlm
G8onZtR92xLSEksGGyAAp19Rx9510Rqo27+qooJTQ0D/0vjyf9i+dn9x3cczpmDs
fN9ucu5Bd9TCNlP84pIw7rEyhA+otly47dgC6uL7digv0lfUtwenOgnNWY0N9Y20
1SLiF5lyFpEmxq9m/bo6Vci2EWgf4D6Ib+R6ZpRUFvBZUHp6rfXl9Qi1bD5qM4It
JavL7XGk7CyGyF0/SCv3Sx0KP1d54poRY/iXeJ5LES0LpAurXS8/yX4i
```

-----END CERTIFICATE-----

```
",
"password": "6ce714ad-6629-4296-a676-1c0ca5f4689c"
},
"id": "95",
"name": "ITC00E0C514325B",
"parentID": "-1",
"movedToBin": false,
"objectType": "tc",
"links": []
}
```

GET /v3/thinclients/{tcId}/assignments/profiles

Summary

Gets the profile and master profile assignments for the specified thin client, in order of their application.



Resource URL

/v3/thinclients/{tcId}/assignments/profiles

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/thinclients/48335/assignments/profiles
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	thin client ID

Response Type

Returns a list of Profile and Master Profile [Assignments](#)(see page 74).

Example Response

```
[
{
  "assignee": {
    "id": "68257",
    "type": "profile"
  },
  "receiver": {
    "id": "48335",
    "type": "tc"
  },
  "assignmentPosition": 0,
  "links": [
    {
      "rel": "assigned",
      "href": "https://172.30.91.227:8443/umsapi/v3/profiles/68257"
    },
    {
```



```
"rel": "receiver",
"href": "https://172.30.91.227:8443/umsapi/v3/thinclients/48335"
},
{
  "rel": "self",
  "href": "https://172.30.91.227:8443/umsapi/v3/profiles/68257/assignments/thinclients/48335"
}
]
},
{
  "assignee": {
    "id": "72063",
    "type": "masterprofile"
  },
  "receiver": {
    "id": "48335",
    "type": "tc"
  },
  "assignmentPosition": 1,
  "links": [
    {
      "rel": "assigned",
      "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72063"
    },
    {
      "rel": "receiver",
      "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/48335"
    },
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72063/assignments/thinclients/48335"
    }
  ]
}
```



```
}  
]
```

PUT /v3/thinclients/{tcId}/assignments/profiles

Summary

Creates Profile and Master Profile Assignments for the specified Thin Client.

Resource URL

/v3/thinclients/{tcId}/assignments/profiles

Example Request

```
curl \  
--request PUT \  
--data ' [{ "assignee": {  
  "id": "68257",  
  "type": "profile"  
},  
  "receiver": {  
    "id": "48335",  
    "type": "tc"  
  }  
}]' \  
https://[server]:8443/umsapi/v3/thinclients/48335/assignments/profiles
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	thin client ID

Request Body

A list of Profile and Master Profile [Assignments](#)(see page 74).

Example Response

```
{  
  "message": "1 assignments successfully assigned to thinclient <48335>"  
}
```



PUT /v3/thinclients

Summary

Creates a new thin client resource.

Resource URL

/v3/thinclients

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '{"mac":"00E0C53C3881", \
"firmwareID":"2"}' \
https://[server]:8443/umsapi/v3/thinclients/
```

Request Body

Name	Type	Mandatory	Description
mac	String (12)	yes	thin client MAC adress
firmwareID	String	yes	firmware ID
name	String	no	thin client name
parentID	String	no	ID of parent directory
site	String	no	thin client site
department	String	no	department
costCenter	String	no	cost center
lastIP	String	no	last known IP of thin client
comment	String	no	comment field
assetID	String	no	asset ID



inServiceDate	String	no	
serialNumber	String	no	serial number of thin client

Response Body

Name	Type	Description
message	String	success or error message
id	String	thin client ID
parentID	String	ID of parent directory
name	String	thin client name

Example Response

```
{
  "message": "Thin client successfully inserted.",
  "id": "7735",
  "name": "My Name",
  "parentID": "-1"
}
```

PUT /v3/thinclients/{tclid}

Summary

Updates properties of the specified thin client

Resource URL

/v3/thinclients/{tclid}

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
```



```
--data '{"name":"reception thin client", \
"site":"main campus"}' \
https://[server]:8443/umsapi/v3/thinclients/123
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	thin client ID

Request Body

Name	Type	Mandatory	Description
name	String	no	thin client name
site	String	no	site of operation
department	String	no	department
costCenter	String	no	cost center
lastIP	String	no	last known IP of thin client
comment	String	no	comment field
assetID	String	no	asset ID
inServiceDate	String	no	
serialNumber	String	no	serial number of thin client

Response Type

Returns a success message.

Example Response

```
{
```




```
"message": "Update successful"
}
```

DELETE /v3/thinclients/{tclid}

Summary

Deletes a thin client.

Warning: This does not move the thin client into the **Recycle Bin** but simply deletes it.

Resource URL

/v1/thinclients/[id]

Example Request

```
curl \
--request DELETE \
https://[server]:8443/umsapi/v3/thinclients/2704
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	thin client ID

Response Type

Returns a message.

Example Response

```
200 OK
---
{
  "CommandExecList": [
    {
      "execID": "ID-PM-MH-WIN7-UMS-63885-1424682219085-5-0",
      "mac": "008064AD82FB",
      "message": "OK",
      "exectime": 1424698605821,
```



```
"state": "SUCCESS"
}
]
}
```

DELETE /v3/thinclients/{tclid}/deletetcoffline

Summary

Deletes a thin client even if it is offline.

This does not move the thin client into the **Recycle Bin** but simply deletes it.

Resource URL

/v3/thinclients/[id]/deletetcoffline

Example Request

```
curl \
--request DELETE \
https://[server]:8443/umsapi/v3/thinclients/2704/deletetcoffline
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	thin client ID

Response Type

Returns a message.

Example Response

```
200 OK
---
{
  "message": "Offline deletion successful"
}
```



POST /v3/thinclients/?command={reboot|shutdown|wakeup}

Summary

Sends a command to all thin clients listed in the request body.

Resource URL

/v3/thinclients/?command={reboot|shutdown|wakeup}

Example Request

```
curl \
--request POST \
--data '[{"id":"27", "type":"tc"}, {"id":"72014", "type":"tc"}]'
https://[server]:8443/umsapi/v3/thinclients?command=wakeup
```

Request Body

A list of [APIObjects](#) (see page 67).

Example Response

```
{
  "CommandExecList": [
    {
      "execID": "ID-PM-MH-WIN7-UMS-54530-1456839861871-5-0",
      "id": "72014",
      "mac": "00E0C561EEED",
      "exectime": "1456845240566",
      "message": "OK",
      "state": "SUCCESS"
    },
    {
      "execID": "ID-PM-MH-WIN7-UMS-54530-1456839861871-5-0",
      "id": "27",
      "mac": "00E0C54DCB8E",
      "exectime": "1456845240560",
      "message": "OK",
      "state": "SUCCESS"
    }
  ]
}
```



```
}
]
}
```

POST /v3/thinclients?command=settings2tc

Summary

Sends settings modified in the UMS database to all thin clients listed in the request body immediately.

Resource URL

/v3/thinclients/?command=settings2tc

Example Request

```
curl \
--request POST \
--data '[{"id":"27", "type":"tc"}, {"id":"72014", "type":"tc"}]'
https://[server]:8443/umsapi/v3/thinclients?command=settings2tc
```

Request Body

A list of [APIObjects](#) (see page 67).

Example Response

```
{
  "CommandExecList": [
    {
      "execID": "ID-PM-MH-WIN7-UMS-54530-1456839861871-5-0",
      "id": "72014",
      "mac": "00E0C561EEED",
      "exectime": "1456845240566",
      "message": "OK",
      "state": "SUCCESS"
    },
    {
      "execID": "ID-PM-MH-WIN7-UMS-54530-1456839861871-5-0",
      "id": "27",
      "mac": "00E0C54DCB8E",
```



```
"exectime": "1456845240560",  
"message": "OK",  
"state": "SUCCESS"  
}  
]  
}
```

POST /v3/thinclients/?command=tcreset2facdefs

Summary

Resets all thin clients listed in the request body to factory defaults.

Resource URL

/v3/thinclients/?command=tcreset2facdefs

Example Request

```
curl \  
--request POST \  
--data ' [{"id": "71342", "type": "tc"} ] '  
https://[server]:8443/umsapi/v3/thinclients/?command=tcreset2facdefs
```

Request Body

A list of [APIObjects](#)(see [page 67](#)).

Example Response

```
{  
  "CommandExecList": [  
    {  
      "execID": "ID-mhuber-59383-1501855717835-6-0",  
      "id": "71342",  
      "exectime": "1501855812301",  
      "message": "OK",  
      "state": "SUCCESS"  
    }  
  ]  
}
```



3.6.3 Profile

- [GET /v3/profiles](#)(see page 118)
- [GET /v3/profiles/{profileid}](#)(see page 120)
- [PUT /v3/profiles/{profileid}](#)(see page 121)
- [DELETE /v3/profiles/{profileid}](#)(see page 122)
- [GET /v3/profiles/{profileid}/assignments/thinclients](#)(see page 123)
- [PUT /v3/profiles/{profileid}/assignments/thinclients/](#)(see page 125)
- [DELETE /v3/profiles/{profileid}/assignments/thinclients/{id}](#)(see page 126)
- [GET /v3/profiles/{profileid}/assignments/tcdirectories](#)(see page 127)
- [PUT /v3/profiles/{profileid}/assignments/tcdirectories](#)(see page 129)
- [DELETE /v3/profiles/{profileid}/assignments/tcdirectories/{id}](#)(see page 130)

GET /v3/profiles

Summary

Gets information on all profiles on the UMS instance.

This method will also list profiles that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/profiles/

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v2/profiles
```

Response Type

Returns a list of profiles.

Example Response

```
[
{
  "firmwareID": "2",
```



```
"isMasterProfile": false,
"overridesSessions": false,
"id": "6585",
"name": "Benchmark intern",
"parentID": "20453",
"movedToBin": false,
"objectType": "profile",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/profiles/6585"
},
{
"rel": "Parent",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20453"
},
{
"rel": "Firmware",
"href": "https://172.30.91.227:8443/umsapi/v3/firmwares/2"
}
],
{
"firmwareID": "2",
"isMasterProfile": false,
"overridesSessions": false,
"id": "6592",
"name": "Benchmark extern",
"parentID": "20453",
"movedToBin": true,
"objectType": "profile",
"links": [
{
"rel": "self",
```



```
"href": "https://172.30.91.227:8443/umsapi/v3/profiles/6592"
},
{
  "rel": "Parent",
  "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20453"
},
{
  "rel": "Firmware",
  "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/2"
}
]
},
[...]
```

GET /v3/profiles/{profileid}

Summary

Gets information on the specified profile.

Resource URL

/v3/profiles/{profileid}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/profiles/6585
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID

Response Type

Returns a profile.



Example Response

```
{
  "firmwareID": "2",
  "isMasterProfile": false,
  "overridesSessions": false,
  "id": "6585",
  "name": "Benchmark intern",
  "parentID": "20453",
  "movedToBin": false,
  "objectType": "profile",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/profiles/6585"
    },
    {
      "rel": "Parent",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20453"
    },
    {
      "rel": "Firmware",
      "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/2"
    }
  ]
}
```

PUT /v3/profiles/{profileid}

Summary

Updates a profile name.

Resource URL

/v3/profiles/{profileid}



Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '{"name":"New Profile Name"}' \
https://[server]:8443/umsapi/v3/profiles/6585
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID

Request Body

Name	Type	Mandatory	Description
name	String	no	profile name

Response Type

Returns a success message.

Example Response

```
{
  "message": "Update successful"
}
```

DELETE /v3/profiles/{profileid}

Summary

Deletes profile.

This does not move the master profile into the **Recycle Bin** but simply deletes it.

Resource URL

/v3/profiles/{profileid}



Example Request

```
curl \
--request DELETE \
https://[server]:8443/umsapi/v3/profiles/72372
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	profile ID

Response Type

Returns a message.

Example Response

```
{
  "message": "Deleted profile with id 72372"
}
```

GET /v3/profiles/{profileid}/assignments/thinclients

Summary

Gets the thin clients the profile is assigned.to.

Resource URL

/v3/profile/{profileid}/assignments/thinclients

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/profiles/20467/assignments/thinclients
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID



Response Type

Returns a list of [Assignments](#)(see page 74).

Example Response

```
[
  {
    "assignee": {
      "id": "20467",
      "type": "profile"
    },
    "receiver": {
      "id": "23028",
      "type": "tc"
    },
    "assignmentPosition": 0,
    "links": [
      {
        "rel": "assigned",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20467"
      },
      {
        "rel": "receiver",
        "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/23028"
      },
      {
        "rel": "self",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20467/assignments/thinclients/23028"
      }
    ]
  },
  {
    "assignee": {
      "id": "20467",
```



```
"type": "profile"
},
"receiver": {
  "id": "48335",
  "type": "tc"
},
"assignmentPosition": 1,
"links": [
  {
    "rel": "assigned",
    "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20467"
  },
  {
    "rel": "receiver",
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/48335"
  },
  {
    "rel": "self",
    "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20467/assignments/thinclients/48335"
  }
]
}
```

PUT /v3/profiles/{profileid}/assignments/thinclients/

Summary

Assigns a profile to one or more thin clients.

Resource URL

/v3/profiles/{profileid}/assignments/thinclients/

Example Request

```
curl \
```



```
--request PUT \  
--header "Content-type: application/json" \  
--data ' [{ "assignee": {  
  "id": "20452",  
  "type": "profile"  
},  
  "receiver": {  
    "id": "23028",  
    "type": "tc"}  
}]' \  
https://[server]:8443/umsapi/v3/profiles/20452/assignments/thinclients/
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID

Request Body

A list of [Assignments](#)(see page 74)

Response Type

Returns a success message.

Example Response

```
{  
  "message": "1 assignments successfully assigned to thinclient <23028>"  
}
```

DELETE /v3/profiles/{profileid}/assignments/thinclients/{id}

Summary

Deletes assignment of the specified profile to the specified thin client.

Resource URL

/v3/profiles/{profileid}/assignments/thinclients/{id}



Example Request

```
curl \
--request DELETE\
https://[server]:8443/umsapi/v3/profiles/68257/assignments/thinclients/48335
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID
id	String	yes	thin client ID

Response Type

Returns a message.

Example Response

```
200 OK
---
{
  "message": "deleted profile assignment"
}
```

GET /v3/profiles/{profileid}/assignments/tcdirectories

Summary

Gets the thin clients the profile is assigned to.

Resource URL

/v3/profile/{profileid}/assignments/thinclients

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/profiles/35549/assignments/tcdirectories/76462
```



Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID

Response Type

Returns a list of [Assignments](#)(see page 74).

Example Response

```
[
  {
    "assignee": {
      "id": "35549",
      "type": "profile"
    },
    "receiver": {
      "id": "76462",
      "type": "tcdirectory"
    },
    "assignmentPosition": 0,
    "links": [
      {
        "rel": "assigned",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/35549"
      },
      {
        "rel": "receiver",
        "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76462"
      },
      {
        "rel": "self",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/35549/assignments/tcdirectories/76462"
      }
    ]
  }
]
```




```
]
}
]
```

PUT /v3/profiles/{profileid}/assignments/tcdirectories

Summary

Assigns a profile to one or more thin client directories.

Resource URL

/v3/profiles/{profileid}/assignments/tcdirectories/

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data ' [{ "assignee": {
  "id": "35549",
  "type": "profile"
},
"receiver": {
  "id": "76462",
  "type": "tcdirectory"}
}]' \
https://[server]:8443/umsapi/v3/profiles/35549/assignments/tcdirectories/
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID

Request Body

A list of [Assignments](#)(see [page 74](#))

Response Type

Returns a success message.



Example Response

```
{  
  "message": "1 assignments successfully assigned to thinclient directory <35549>"  
}
```

DELETE /v3/profiles/{profileid}/assignments/tcdirectories/{id}

Summary

Deletes assignment of the specified profile to the specified thin client directory.

Resource URL

/v3/profiles/{profileid}/assignments/tcdirectory/{id}

Example Request

```
curl \  
--request DELETE\  
https://[server]:8443/umsapi/v3/profiles/35549/assignments/tcdirectories/76462
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID
id	String	yes	thin client directory ID

Response Type

Returns a message.

Example Response

```
200 OK  
---  
{  
  "message": "deleted profile assignment"
```



```
}
```

3.6.4 Master Profile

- [GET /v3/masterprofiles](#)(see page 131)
- [GET /v3/masterprofiles/{profileid}](#)(see page 133)
- [PUT /v3/masterprofile/{profileid}](#)(see page 134)
- [DELETE /v3/masterprofiles/{profileid}](#)(see page 135)
- [GET /v3/masterprofiles/{profileid}/assignments/thinclients](#)(see page 136)
- [PUT /v3/masterprofiles/{profileid}/assignments/thinclients/](#)(see page 138)
- [DELETE /v3/masterprofiles/{profileid}/assignments/thinclients/{id}](#)(see page 140)
- [GET /v3/masterprofiles/{profileid}/assignments/tcdirectories](#)(see page 140)
- [PUT /v3/masterprofiles/{profileid}/assignments/tcdirectories/](#)(see page 142)
- [DELETE /v3/masterprofiles/{profileid}/assignments/tcdirectories/{id}](#)(see page 143)

GET /v3/masterprofiles

Summary

Gets information on all master profiles on the UMS instance.

This method will also list master profiles that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/masterprofiles/

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/masterprofiles
```

Response Type

Returns a list of profiles.

Example Response

```
[
{
  "firmwareID": "32",
```



```
"isMasterProfile": true,
"overridesSessions": false,
"id": "72054",
"name": "Company Master Profile",
"parentID": "-14",
"movedToBin": false,
"objectType": "masterprofile",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72054"
},
{
"rel": "Parent",
"href": "root"
},
{
"rel": "Firmware",
"href": "https://172.30.91.227:8443/umsapi/v3/firmwares/32"
}
],
{
"firmwareID": "23",
"isMasterProfile": true,
"overridesSessions": false,
"id": "72055",
"name": "Spciaal Master Profile",
"parentID": "-14",
"movedToBin": false,
"objectType": "masterprofile",
"links": [
{
"rel": "self",
```



```
"href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72055"
},
{
  "rel": "Parent",
  "href": "root"
},
{
  "rel": "Firmware",
  "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/23"
}
]
}
```

GET /v3/masterprofiles/{profileid}

Summary

Gets information on the specified master profile.

Resource URL

/v3/masterprofiles/{profileid}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/profiles/72054
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID

Response Type

Returns a master profile.



Example Response

```
{
  "firmwareID": "32",
  "isMasterProfile": true,
  "overridesSessions": false,
  "id": "72054",
  "name": "Company Master Profile",
  "parentID": "-14",
  "movedToBin": false,
  "objectType": "masterprofile",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72054"
    },
    {
      "rel": "Parent",
      "href": "root"
    },
    {
      "rel": "Firmware",
      "href": "https://172.30.91.227:8443/umsapi/v3/firmwares/32"
    }
  ]
}
```

PUT /v3/masterprofile/{profileid}

Summary

Updates a master profile name.

Resource URL

/v3/masterprofiles/{profileid}



Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '{"name":"New Master Profile Name"}' \
https://[server]:8443/umsapi/v3/masterprofiles/6585
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID

Request Body

Name	Type	Mandatory	Description
name	String	no	master profile name

Response Type

Returns a success message.

Example Response

```
{
  "message": "Update successful"
}
```

DELETE /v3/masterprofiles/{profileid}

Summary

Deletes a master profile.

This does not move the master profile into the **Recycle Bin** but simply deletes it.



Resource URL

/v3/masterprofiles/{profileid}

Example Request

```
curl \
--request DELETE \
https://[server]:8443/umsapi/v3/masterprofiles/72303
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	master profile ID

Response Type

Returns a message.

Example Response

```
{
  "message": "Deleted profile with id 72303"
}
```

GET /v3/masterprofiles/{profileid}/assignments/thinclients

Summary

Gets the thin clients the master profile is assigned.to.

Resource URL

/v3/masterprofiles/[profileid]/assignments/thinclients

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/masterprofiles/72308/assignments/thinclients
```




Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	profile ID

Response Type

Returns a list of [Assignments](#)(see page 74).

Example Response

```
[
  {
    "assignee": {
      "id": "72308",
      "type": "masterprofile"
    },
    "receiver": {
      "id": "48335",
      "type": "tc"
    },
    "assignmentPosition": 0,
    "links": [
      {
        "rel": "assigned",
        "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72308"
      },
      {
        "rel": "receiver",
        "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/48335"
      },
      {
        "rel": "self",
        "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72308/assignments/thinclients/48335"
      }
    ]
  }
]
```



```

]
},
{
  "assignee": {
    "id": "72308",
    "type": "masterprofile"
  },
  "receiver": {
    "id": "72014",
    "type": "tc"
  },
  "assignmentPosition": 1,
  "links": [
    {
      "rel": "assigned",
      "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72308"
    },
    {
      "rel": "receiver",
      "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/72014"
    },
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72308/assignments/
thinclients/72014"
    }
  ]
}
]

```

PUT /v3/masterprofiles/{profileid}/assignments/thinclients/

Summary

Assigns a master profile to one or more thin clients.



Resource URL

/v3/masterprofiles/[profileid/assignments/thinclients/]

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data ' [{ "assignee": {
  "id": "72066",
  "type": "masterprofile"
},
  "receiver": {
    "id": "48335",
    "type": "tc"}
}]' \
https://[server]:8443/umsapi/v3/masterprofiles/72066/assignments/thinclients/
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID

Request Body

A list of [Assignments](#)(see page 74)

Response Type

Returns a success message.

Example Response

```
{
  "message": "1 assignments successfully assigned to thinclient <48335>"
}
```



DELETE /v3/masterprofiles/{profileid}/assignments/thinclients/{id}

Summary

Deletes assignment of the specified master profile to the specified thin client.

Resource URL

/v3/masterprofiles/{profileid}/assignments/thinclients/{id}

Example Request

```
curl \
--request DELETE\
https://[server]:8443/umsapi/v3/masterprofiles/68257/assignments/thinclients/48335
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID
id	String	yes	thin client ID

Response Type

Returns a message.

Example Response

```
200 OK
---
{
  "message": "deleted profile assignment"
}
```

GET /v3/masterprofiles/{profileid}/assignments/tcdirectories

Summary

Gets the thin client directories the master profile is assigned to.



Resource URL

/v3/masterprofiles/{profileid}/assignments/tcdirectories

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/masterprofiles/72098/assignments/tcdirectories
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	Master profile ID

Response Type

Returns a list of [Assignments](#)(see page 74).

Example Response

```
[
{
  "assignee": {
    "id": "72098",
    "type": "masterprofile"
  },
  "receiver": {
    "id": "76462",
    "type": "tcdirectory"
  },
  "assignmentPosition": 0,
  "links": [
    {
      "rel": "assigned",
      "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72098"
    },
    {
```



```

    "rel": "receiver",
    "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76462"
  },
  {
    "rel": "self",
    "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72098/assignments/
tcdirectories/76462"
  }
]
}
]

```

PUT /v3/masterprofiles/{profileid}/assignments/tcdirectories/

Summary

Assigns a master profile to one or more thin client directory.

Resource URL

/v3/masterprofiles/{profileid}/assignments/tcdirectories/]

Example Request

```

curl \
--request PUT \
--header "Content-type: application/json" \
--data ' [{ "assignee": {
  "id": "72098",
  "type": "masterprofile"
},
  "receiver": {
    "id": "76462",
    "type": "tcdirectory"}
}]' \
https://[server]:8443/umsapi/v3/masterprofiles/72066/assignments/thinclients/

```



Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID

Request Body

A list of [Assignments](#)(see page 74)

Response Type

Returns a success message.

Example Response

```
{  
  "message": "1 assignments successfully assigned to thinclient directory <72098>"  
}
```

DELETE /v3/masterprofiles/{profileid}/assignments/tcdirectories/{id}

Summary

Deletes assignment of the specified master profile to the specified thin client directory.

Resource URL

/v3/masterprofiles/{profileid}/assignments/tcdirectory/{id}

Example Request

```
curl \  
--request DELETE\  
https://[server]:8443/umsapi/v3/masterprofiles/72098/assignments/tcdirectories/76462
```

Request Path Variables

Name	Type	Mandatory	Description
profileid	String	yes	master profile ID
id	String	yes	thin client directory ID



Response Type

Returns a message.

Example Response

200 OK

```
{  
  "message": "deleted profile assignment"  
}
```

3.6.5 Thin Client Directory

- [GET /v3/directories/tcdirectories](#)(see page 144)
- [GET /v3/directories/tcdirectories?facets=children](#)(see page 146)
- [GET /v3/directories/tcdirectories/{id}](#)(see page 149)
- [GET /v3/directories/tcdirectories/{id}?facets=children](#)(see page 150)
- [PUT /v3/directories/tcdirectories/](#)(see page 153)
- [PUT /v3/directories/tcdirectories/{id}](#)(see page 154)
- [PUT /v3/directories/tcdirectories/{id}?operation=move](#)(see page 155)
- [DELETE /v3/directories/tcdirectories/{id}](#)(see page 156)
- [GET /v3/directories/tcdirectories/{id}/assignments/profiles](#)(see page 157)
- [PUT /v3/directories/tcdirectories/{id}/assignments/profiles](#)(see page 159)

GET /v3/directories/tcdirectories

Summary

Gets information on all Thin Client Directories in a flat format.

This method will also list directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/tcdirectories

Example Request

```
curl \  
--request GET \  

```




`https://[server]:8443/umsapi/v3/directories/tcdirectories`

Response Type

Returns a list of [Directories](#)(see page 76).

Example Response

```
[
  {
    "id": "50705",
    "name": "New",
    "parentID": "-1",
    "movedToBin": false,
    "objectType": "tcdirectory",
    "links": [
      {
        "rel": "self",
        "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/50705"
      },
      {
        "rel": "Parent",
        "href": "root"
      }
    ]
  },
  {
    "id": "15592",
    "name": "Pool",
    "parentID": "-1",
    "movedToBin": false,
    "objectType": "tcdirectory",
    "links": [
      {
        "rel": "self",
        "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/15592"
      }
    ]
  }
]
```



```
{  
  "rel": "Parent",  
  "href": "root"  
}  
]  
}
```

GET /v3/directories/tcdirectories?facets=children

Summary

Gets information on all Thin Client Directories, recursively listing their children.

This method will also list directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/tcdirectories?facets=children

Example Request

```
curl \  
--request GET \  
https://[server]:8443/umsapi/v3/directories/tcdirectories/facets=children
```

Response Type

Returns a list of [Directories](#)(see page 76), with their DirectoryChildren fields containing other objects.

Example Response

```
[  
  {  
    "id": "50705",  
    "name": "New",  
    "parentID": "-1",  
    "movedToBin": true,  
    "objectType": "tcdirectory",  
    "links": [  
      {
```



```
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/50705"
},
{
  "rel": "Parent",
  "href": "root"
}
],
{
  "DirectoryChildren": [
    {
      "objectType": "tc",
      "id": "7117",
      "link": {
        "rel": "tc",
        "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/7117"
      }
    },
    {
      "objectType": "tc",
      "id": "6888",
      "link": {
        "rel": "tc",
        "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/6888"
      }
    },
    {
      "objectType": "tc",
      "id": "23028",
      "link": {
        "rel": "tc",
        "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/23028"
      }
    }
  ]
}
```



```
}
],
"id": "72327",
"name": "Sub",
"parentID": "15592",
"movedToBin": false,
"objectType": "tcdirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/72327"
},
{
"rel": "Parent",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/15592"
}
],
{
"DirectoryChildren": [
{
"objectType": "tcdirectory",
"id": "72327",
"link": {
"rel": "tcdirectory",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/72327"
}
},
{
"objectType": "tc",
"id": "7121",
"link": {
"rel": "tc",
"href": "https://172.30.91.227:8443/umsapi/v3/thinclients/7121"
}
```



```
}  
},  
{  
  "objectType": "tc",  
  "id": "6907",  
  "link": {  
    "rel": "tc",  
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/6907"  
  }  
},  
{  
[...]
```

GET /v3/directories/tcdirectories/{id}

Summary

Gets information on a specific Thin Client Directory.

This method will also list directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/tcdirectories/{id}

Example Request

```
curl \  
--request GET \  
https://[server]:8443/umsapi/v3/directories/tcdirectories/15592
```

Response Type

Returns a [Directory](#) (see page 76).

Example Response

```
{  
  "id": "15592",  
  "name": "Pool",  
  "parentID": "-1",
```



```
"movedToBin": false,
"objectType": "tcdirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/15592"
},
{
"rel": "Parent",
"href": "root"
}
]
```

GET /v3/directories/tcdirectories/{id}?facets=children

Summary

Gets information a Thin Client Directory, recursively listing its children.

This method will also list directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/tcdirectories/{id}?facets=children

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/tcdirectories/15592?facets=children
```

Response Type

Returns a [Directory](#) (see page 76), with its DirectoryChildren field containing other objects.

Example Response

```
{
  "DirectoryChildren": [
```



```
{
  "objectType": "tcdirectory",
  "id": "72327",
  "link": {
    "rel": "tcdirectory",
    "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/72327"
  }
},
{
  "objectType": "tc",
  "id": "7121",
  "link": {
    "rel": "tc",
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/7121"
  }
},
{
  "objectType": "tc",
  "id": "6907",
  "link": {
    "rel": "tc",
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/6907"
  }
},
{
  "objectType": "tc",
  "id": "11219",
  "link": {
    "rel": "tc",
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/11219"
  }
},
{
  "objectType": "tc",
```



```
"id": "15223",
"link": {
  "rel": "tc",
  "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/15223"
},
{
  "objectType": "tc",
  "id": "15553",
  "link": {
    "rel": "tc",
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/15553"
  },
  {
    "objectType": "tc",
    "id": "7113",
    "link": {
      "rel": "tc",
      "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/7113"
    },
    {
      "objectType": "tc",
      "id": "6899",
      "link": {
        "rel": "tc",
        "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/6899"
      },
      {
        "objectType": "tc",
        "id": "6912",
        "link": {
```




```

"rel": "tc",
"href": "https://172.30.91.227:8443/umsapi/v3/thinclients/6912"
},
{
  "objectType": "tc",
  "id": "26836",
  "link": {
    "rel": "tc",
    "href": "https://172.30.91.227:8443/umsapi/v3/thinclients/26836"
  }
},
{
  "id": "15592",
  "name": "Pool",
  "parentID": "-1",
  "movedToBin": false,
  "objectType": "tcdirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/15592"
    },
    {
      "rel": "Parent",
      "href": "root"
    }
  ]
}

```

PUT /v3/directories/tcdirectories/

Summary

Creates a Thin Client Directory.



Resource URL

/v3/directories/tcdirectories/

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '{"name":"Test"}' \
https://[server]:8443/umsapi/v3/directories/tcdirectories/
```

Request Body

Name	Type	Mandatory	Description
name	String	no	directory name

Response Type

Returns a success message.

Example Response

```
{
  "message": "Directory successfully inserted.",
  "id": "72340",
  "name": "Test"
}
```

PUT /v3/directories/tcdirectories/{id}

Summary

Updates a Thin Client Directory name.

Resource URL

/v3/directories/tcdirectories/{id}

Example Request

```
curl \
```



```
--request PUT \  
--header "Content-type: application/json" \  
--data '{"name":"New Directory Name"}' \  
https://[server]:8443/umsapi/v3/directories/tcdirectories/72340
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	directory ID

Request Body

Name	Type	Mandatory	Description
name	String	no	directory name

Response Type

Returns a success message.

Example Response

```
{  
  "message": "Updated directory successfully."  
}
```

PUT /v3/directories/tcdirectories/{id}?operation=move

Summary

Moves Thin Clients and Thin Client Directories into the specified Thin Client Directory.

Resource URL

/v3/directories/tcdirectories/{id}?operation=move

Example Request

```
curl \  
--request PUT \  
--header "Content-type: application/json" \  
https://[server]:8443/umsapi/v3/directories/tcdirectories/{id}?operation=move
```



```
--data '[{"id": "76462", "type": "tcdirectory"}, {"id": "7121", "type": "tc"}]' \
https://[server]:8443/umsapi/v3/directories/tcdirectories/72340?operation=move
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	Thin Client Directory ID

Request Body

A list of [ApiObjects](#)(see page 67)

Response Type

Returns a success message.

Example Response

```
[
  {
    "id": "7121",
    "results": "successful"
  },
  {
    "id": "76462",
    "results": "successful"
  }
]
```

DELETE /v3/directories/tcdirectories/{id}

Summary

Deletes a Thin Client Directory.

This deletes only empty directories. The attempt to delete a non-empty directory results in an error.

**Resource URL**

/v3/directories/tcdirectories/{id}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/tcdirectories/72327
```

Response Type

Returns a success message.

Example Response

```
{
  "message": "Deletion successful."
}
```

GET /v3/directories/tcdirectories/{id}/assignments/profiles**Summary**

Gets the profile and master profile assignments for the specified thin client directory, in order of their application.

Resource URL

/v3/directories/tcdirectories/{id}/assignments/profiles

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/tcdirectories/76462/assignments/profiles
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	Thin Client Directory ID



Response Type

Returns a list of Profile and Master Profile [Assignments](#)(see page 74).

Example Response

```
[
  {
    "assignee": {
      "id": "35549",
      "type": "profile"
    },
    "receiver": {
      "id": "76462",
      "type": "tcdirectory"
    },
    "assignmentPosition": 0,
    "links": [
      {
        "rel": "assigned",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/35549"
      },
      {
        "rel": "receiver",
        "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76462"
      },
      {
        "rel": "self",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/35549/assignments/tcdirectories/76462"
      }
    ]
  },
  {
    "assignee": {
      "id": "72098",
```



```

"type": "masterprofile"
},
"receiver": {
  "id": "76462",
  "type": "tcdirectory"
},
"assignmentPosition": 1,
"links": [
  {
    "rel": "assigned",
    "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72098"
  },
  {
    "rel": "receiver",
    "href": "https://172.30.91.227:8443/umsapi/v3/directories/tcdirectories/76462"
  },
  {
    "rel": "self",
    "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72098/assignments/tcdirectories/76462"
  }
]
}
]

```

PUT /v3/directories/tcdirectories/[id]/assignments/profiles

Summary

Creates profile and master profile assignments for the specified thin client directory.

Resource URL

/v3/directories/tcdirectories/{id}/assignments/profiles

Example Request

```
curl \
```



```
--request PUT \
--data '[{ "assignee": {
  "id": "35549",
  "type": "profile"
},
"receiver": {
  "id": "76462",
  "type": "tcdirectory"}
}]' \
```

https://[server]:8443/umsapi/v3/directories/tcdirectories/76462/assignments/profiles

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	Thin Client Directory ID

Request Body

A list of Profile and Master Profile [Assignments](#)(see page 74).

Example Response

```
{
  "message": "1 assignments successfully assigned to thinclient directory <76462>"
}
```

3.6.6 Profile Directory

- [GET /v3/directories/profiledirectories](#)(see page 161)
- [GET /v3/directories/profiledirectories?facets=children](#)(see page 163)
- [GET /v3/directories/profiledirectories{id}](#)(see page 167)
- [GET /v3/directories/profiledirectories\[id\]?facets=children](#)(see page 168)
- [PUT /v3/directories/profiledirectories/](#)(see page 170)
- [PUT /v3/directories/profiledirectories/{id}](#)(see page 171)
- [PUT /v3/directories/profiledirectories/\[id\]?operation=move](#)(see page 172)
- [DELETE /v3/directories/profiledirectories/{id}](#)(see page 173)



GET /v3/directories/profiledirectories

Summary

Gets information on all Profile Directories in a flat format..

This method will also list profile directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/profiledirectories

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/profiledirectories
```

Response Type

Returns a list of Profile Directories.

Example Response

```
[
{
  "id": "72064",
  "name": "New Profile Directory",
  "parentID": "-2",
  "movedToBin": true,
  "objectType": "profiledirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/72064"
    },
    {
      "rel": "Parent",
```



```
"href": "root"
}
],
{
  "id": "20451",
  "name": "Custom Partitions",
  "parentID": "-2",
  "movedToBin": false,
  "objectType": "profiledirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20451"
    },
    {
      "rel": "Parent",
      "href": "root"
    }
  ],
},
{
  "id": "20707",
  "name": "Standards",
  "parentID": "-2",
  "movedToBin": false,
  "objectType": "profiledirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20707"
    },
    {
      "rel": "Parent",
```



```
"href": "root"
}
],
{
  "id": "20453",
  "name": "Benchmarks",
  "parentID": "-2",
  "movedToBin": false,
  "objectType": "profiledirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20453"
    },
    {
      "rel": "Parent",
      "href": "root"
    }
  ]
}
```

GET /v3/directories/profiledirectories?facets=children

Summary

Gets information on all Profile Directories, recursively listing their children.

This method will also list profile directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/profiledirectories?facets=children



Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/profiledirectories/facets=children
```

Response Type

Returns a list of Profile Directories,, with their DirectoryChildren fields containing other objects.

Example Response

```
[
{
  "DirectoryChildren": [
    {
      "objectType": "profile",
      "id": "20467",
      "link": {
        "rel": "profile",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20467"
      }
    },
    {
      "objectType": "profile",
      "id": "20452",
      "link": {
        "rel": "profile",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20452"
      }
    },
    {
      "objectType": "profile",
      "id": "20449",
      "link": {
        "rel": "profile",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20449"
      }
    }
  ]
}
```



```
}
}
],
"id": "20707",
"name": "Standards",
"parentID": "-2",
"movedToBin": false,
"objectType": "profiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20707"
},
{
"rel": "Parent",
"href": "root"
}
],
{
"DirectoryChildren": [
{
"objectType": "profiledirectory",
"id": "72068",
"link": {
"rel": "profiledirectory",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/72068"
}
},
{
"objectType": "profile",
"id": "6592",
"link": {
"rel": "profile",
```



```
"href": "https://172.30.91.227:8443/umsapi/v3/profiles/6592"
}
},
{
  "objectType": "profile",
  "id": "6585",
  "link": {
    "rel": "profile",
    "href": "https://172.30.91.227:8443/umsapi/v3/profiles/6585"
  }
}
],
"id": "20453",
"name": "Benchmarks",
"parentID": "-2",
"movedToBin": false,
"objectType": "profiledirectory",
"links": [
  {
    "rel": "self",
    "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20453"
  },
  {
    "rel": "Parent",
    "href": "root"
  }
]
},
{
  "DirectoryChildren": [],
  "id": "72068",
  "name": "A Subdirectory",
  "parentID": "20453",
  "movedToBin": false,
```



```
"objectType": "profiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/72068"
},
{
"rel": "Parent",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20453"
}
]
}
```

GET /v3/directories/profiledirectories{id}

Summary

Gets information on a specific Profile Directory.

This method will also show directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/profiledirectories

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/profiledirectories/20707
```

Response Type

Returns a Profile Directory.

Example Response

```
{
```



```
"id": "20707",
"name": "Standards",
"parentID": "-2",
"movedToBin": false,
"objectType": "profiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20707"
},
{
"rel": "Parent",
"href": "root"
}
]
```

GET /v3/directories/profiledirectories[id]?facets=children

Summary

Gets information on a specific Profile Directory, recursively listing its children.

This method will also show directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/profiledirectories{id}?facets=children

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/profiledirectories/20707?facets=children
```

Response Type

Returns a Profile Directory, with its DirectoryChildren fields containing other objects



Example Response

```
{
  "DirectoryChildren": [
    {
      "objectType": "profile",
      "id": "20467",
      "link": {
        "rel": "profile",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20467"
      }
    },
    {
      "objectType": "profile",
      "id": "20452",
      "link": {
        "rel": "profile",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20452"
      }
    },
    {
      "objectType": "profile",
      "id": "20449",
      "link": {
        "rel": "profile",
        "href": "https://172.30.91.227:8443/umsapi/v3/profiles/20449"
      }
    }
  ],
  "id": "20707",
  "name": "Standards",
  "parentID": "-2",
  "movedToBin": false,
  "objectType": "profiledirectory",
}
```



```
"links": [  
  {  
    "rel": "self",  
    "href": "https://172.30.91.227:8443/umsapi/v3/directories/profiledirectories/20707"  
  },  
  {  
    "rel": "Parent",  
    "href": "root"  
  }  
]  
}
```

PUT /v3/directories/profiledirectories/

Summary

Creates a Profile Directory.

Resource URL

/v3/directories/profiledirectories/

Example Request

```
curl \  
--request PUT \  
--header "Content-type: application/json" \  
--data '{"name":"New Profiles"}' \  
https://[server]:8443/umsapi/v3/directories/profiledirectories/
```

Request Body

Name	Type	Mandatory	Description
name	String	no	directory name

Response Type

Returns a success message.



Example Response

```
{  
  "message": "Directory successfully inserted.",  
  "id": "72090",  
  "name": "New Profiles"  
}
```

PUT /v3/directories/profiledirectories/{id}

Summary

Updates a Profile Directory name.

Resource URL

/v3/directories/profiledirectories/{id}

Example Request

```
curl \  
--request PUT \  
--header "Content-type: application/json" \  
--data '{"name": "New Directory Name"}' \  
https://[server]:8443/umsapi/v3/directories/profiledirectories/72340
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	directory ID

Request Body

Name	Type	Mandatory	Description
name	String	no	directory name



Response Type

Returns a success message.

Example Response

```
{  
  "message": "Updated directory successfully."  
}
```

PUT /v3/directories/profiledirectories/[id]?operation=move

Summary

Moves Profiles and Profile Directories into the specified Profile Directory.

Resource URL

/v3/directories/profiledirectories/[id]?operation=move

Example Request

```
curl \  
--request PUT \  
--header "Content-type: application/json" \  
--data ' [{"id": "20451", "type": "profiledirectory"},  
{"id": "6592", "type": "profile"} ] ' \  
https://[server]:8443/umsapi/v3/directories/profiledirectories/72068?operation=move
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	directory ID

Request Body

A list of [ApiObjects](#)(see page 67)

Response Type

Returns a success message.



Example Response

```
[  
  {  
    "id": "20451",  
    "results": "successful"  
  },  
  {  
    "id": "6592",  
    "results": "successful"  
  }  
]
```

DELETE /v3/directories/profiledirectories/{id}

Summary

Deletes a Profile Directory.

This deletes only empty directories. The attempt to delete a non-empty directory results in an error.

Resource URL

/v3/directories/profiledirectories/{id}

Example Request

```
curl \  
--request DELETE\  
https://[server]:8443/umsapi/v3/directories/profiledirectories/72327
```

Response Type

Returns a success message.

Example Response

```
{  
  "message": "Deletion successful."  
}
```



3.6.7 Master Profile Directory

- [GET /v3/directories/masterprofiledirectories](#)(see page 174)
- [GET /v3/directories/masterprofiledirectories?facets=children](#)(see page 177)
- [GET /v3/directories/masterprofiledirectories/{id}](#)(see page 179)
- [GET /v3/directories/masterprofiledirectories{id}?facets=children](#)(see page 180)
- [PUT /v3/directories/masterprofiledirectories/](#)(see page 182)
- [PUT /v3/directories/masterprofiledirectories/{id}](#)(see page 183)
- [PUT /v3/directories/masterprofiledirectories/{id}?operation=move](#)(see page 183)
- [DELETE /v3/directories/masterprofiledirectories{id}](#)(see page 185)

GET /v3/directories/masterprofiledirectories

Summary

Gets information on all Master Profile Directories in a flat format..

This method will also list master profile directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/masterprofiledirectories

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories
```

Response Type

Returns a list of Master Profile Directories.

Example Response

```
[
{
  "id": "72101",
  "name": "Basic",
  "parentID": "-14",
```



```
"movedToBin": false,
"objectType": "masterprofiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72101"
},
{
"rel": "Parent",
"href": "root"
}
],
{
"id": "72100",
"name": "Important",
"parentID": "76552",
"movedToBin": false,
"objectType": "masterprofiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72100"
},
{
"rel": "Parent",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/76552"
}
],
{
"id": "76552",
```



```
"name": "New Directory",
"parentID": "-14",
"movedToBin": false,
"objectType": "masterprofiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/76552"
},
{
"rel": "Parent",
"href": "root"
}
],
{
"id": "76556",
"name": "Old Directory",
"parentID": "-14",
"movedToBin": true,
"objectType": "masterprofiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/76556"
},
{
"rel": "Parent",
"href": "root"
}
]
}
```




GET /v3/directories/masterprofiledirectories?facets=children

Summary

Gets information on all Master Profile Directories, recursively listing their children.

This method will also list master profile directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/masterprofiledirectories?facets=children

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/facets=children
```

Response Type

Returns a list of Master Profile Directories, with their DirectoryChildren fields containing other objects.

Example Response

```
[
{
  "DirectoryChildren": [
    {
      "objectType": "masterprofile",
      "id": "72098",
      "link": {
        "rel": "masterprofile",
        "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72098"
      }
    }
  ],
  "id": "72101",
  "name": "Basic",
```



```
"parentID": "-14",
"movedToBin": false,
"objectType": "masterprofiledirectory",
"links": [
{
"rel": "self",
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72101"
},
{
"rel": "Parent",
"href": "root"
}
],
{
"DirectoryChildren": [
{
"objectType": "masterprofile",
"id": "72096",
"link": {
"rel": "masterprofile",
"href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72096"
}
}
],
"id": "72100",
"name": "Important",
"parentID": "-14",
"movedToBin": false,
"objectType": "masterprofiledirectory",
"links": [
{
"rel": "self",
```



```
"href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72100"
},
{
  "rel": "Parent",
  "href": "root"
}
]
}
```

GET /v3/directories/masterprofiledirectories/{id}

Summary

Gets information on a specific Master Profile Directory.

This method will also show master profile directories that are located in the **Recycle Bin** ("movedToBin":"true").

Resource URL

/v3/directories/masterprofiledirectories/{id}

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/20707
```

Response Type

Returns a Master Profile Directory.

Example Response

```
{
  "id": "72101",
  "name": "Basic",
  "parentID": "-14",
```



```
"movedToBin": false,
"objectType": "masterprofiledirectory",
"links": [
{
  "rel": "self",
  "href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72101"
},
{
  "rel": "Parent",
  "href": "root"
}
]
```

GET /v3/directories/masterprofiledirectories{id}?facets=children

Summary

Gets information on a specific Master Profile Directory, recursively listing its children.

This method will also show master profile directories that are located in the **Recycle Bin** ("movedToBin": "true").

Resource URL

/v3/directories/masterprofiledirectories{id}?facets=children

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/72101?
facets=children
```

Response Type

Returns a Master Profile Directory, with its DirectoryChildren fields containing other objects



Example Response

```
{
  "DirectoryChildren": [
    {
      "objectType": "masterprofiledirectory",
      "id": "72112",
      "link": {
        "rel": "masterprofiledirectory",
        "href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72112"
      }
    },
    {
      "objectType": "masterprofile",
      "id": "72098",
      "link": {
        "rel": "masterprofile",
        "href": "https://172.30.91.227:8443/umsapi/v3/masterprofiles/72098"
      }
    }
  ],
  "id": "72101",
  "name": "Basic",
  "parentID": "-14",
  "movedToBin": false,
  "objectType": "masterprofiledirectory",
  "links": [
    {
      "rel": "self",
      "href": "https://172.30.91.227:8443/umsapi/v3/directories/masterprofiledirectories/72101"
    },
    {
      "rel": "Parent",
```



```
"href": "root"
}
]
}
```

PUT /v3/directories/masterprofiledirectories/

Summary

Creates a Master Profile Directory.

Resource URL

/v3/directories/masterprofiledirectories/

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '{"name":"Yet Another Directory"}' \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/
```

Request Body

Name	Type	Mandatory	Description
name	String	yes	Master profile directory name

Response Type

Returns a success message.

Example Response

```
{
  "message": "Directory successfully inserted.",
  "id": "76560",
  "name": "Yet Another Directory"
}
```



PUT /v3/directories/masterprofiledirectories/{id}

Summary

Renames a Master Profile Directory.

Resource URL

/v3/directories/masterprofiledirectories/{id}

Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '{"name":"New Master Profiles"}' \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/76560
```

Request Body

Name	Type	Mandatory	Description
name	String	yes	Master profile directory name

Response Type

Returns a success message.

Example Response

```
{
  "message": "Updated directory successfully."
}
```

PUT /v3/directories/masterprofiledirectories/{id}?operation=move

Summary

Moves Master Profiles and Master Profile Directories into the specified Master Profile Directory.

Resource URL

/v3/directories/masterprofiledirectories/{id}?operation=move



Example Request

```
curl \
--request PUT \
--header "Content-type: application/json" \
--data '[{"id": "72100", "type": "masterprofiledirectory"}, {"id": "72098",
"type": "masterprofile"}]' \
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/76552?
operation=move
```

Request Path Variables

Name	Type	Mandatory	Description
id	String	yes	directory ID

Request Body

A list of [ApiObjects](#)(see page 67)

Response Type

Returns a success message.

Example Response

```
[
{
  "id": "72098",
  "results": "successful"
},
{
  "id": "72100",
  "results": "successful"
}
]
```




DELETE /v3/directories/masterprofiledirectories{id}

Summary

Deletes a Master Profile Directory.

This deletes only empty directories. The attempt to delete a non-empty directory results in an error.

Resource URL

/v3/directories/masterprofiledirectories/{id}

Example Request

```
curl \
--request DELETE\
https://[server]:8443/umsapi/v3/directories/masterprofiledirectories/76559
```

Response Type

Returns a success message.

Example Response

```
{
  "message": "Deletion successful."
}
```

3.6.8 Firmware Information

- [GET /v3/firmwares](#)(see page 185)
- [GET /v3/firmwares/{fwId}](#)(see page 187)

GET /v3/firmwares

Summary

Gets information on all firmwares known to the UMS.

Resource URL

/v3/firmwares/



Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/firmwares/
```

Response Type

Returns a list of firmwares.

Example Response

```
{
  "FwResource": [
    {
      "id": "2",
      "product": "IGEL Universal Desktop LX",
      "version": "10.02.100.01",
      "firmwareType": "LX",
      "links": []
    },
    {
      "id": "3",
      "product": "IGEL Universal Desktop OS 3",
      "version": "10.02.100.01",
      "firmwareType": "LX",
      "links": []
    },
    {
      "id": "4",
      "product": "IGEL Universal Desktop LX",
      "version": "10.02.120.01",
      "firmwareType": "LX",
      "links": []
    },
    {
      "id": "6",
```



```
"product": "IGEL Universal Desktop LX",
"version": "10.03.100.rc2.01",
"firmwareType": "LX",
"links": []
}
]
```

GET /v3/firmwares/{fwId}

Summary

Gets information on a specific firmware.

Resource URL

/v3/firmwares/[fwId]

Example Request

```
curl \
--request GET \
https://[server]:8443/umsapi/v3/firmwares/4
```

Request Path Variables

Name	Type	Mandatory	Description
fwId	String	yes	firmware ID

Response Type

Returns a firmware.

Example Response

```
{
  "id": "6",
  "product": "IGEL Universal Desktop LX",
  "version": "10.03.100.rc2.01",
  "firmwareType": "LX",
```



```
"links": []  
}
```

3.6.9 Server Information

- [GET /v3/serverstatus](#)(see page 188)

GET /v3/serverstatus

Summary

Gets diagnostic information about the UMS instance.

Server status is the only resource that can be queried without logging in. This makes it useful for debugging the connection to the IMI service.

Resource URL

/v3/serverstatus

Example Request

```
curl \  
--request GET \  
https://[server]:8443/umsapi/v3/serverstatus
```

Example Response

```
{  
  "rmGuiServerVersion": "5.07.100.rc10",  
  "buildNumber": "31933",  
  "activeMQVersion": "5.6.0",  
  "derbyVersion": "10.12.1.1",  
  "serverUUID": "9fe719f1-c16e-4744-9ff1-b9c314ae151c",  
  "server": "mhuber:8443",  
  "links": []  
}
```



```
}
```

3.7 Error Codes

- [HTTP Status Codes](#)(see page 189)
- [API Error Messages](#)(see page 189)

3.7.1 HTTP Status Codes

Client-Side Errors

HTTP Status	Title	Description
200	OK	The request was successful. However, it is a good idea to read the API message.
400	Bad Request	The request does not match the API, e.g. it uses the wrong HTTP method
401	Unauthorized	The client has not logged in or has sent the wrong credentials.
404	Not Found	The endpoint does not exist, it may be misspelled.
415	Unsupported Media Type	The body content, e.g. JSON, does not match the Content-Type header or is not well-formed.

Server-Side Errors

HTTP Status	Title	Description
500	Internal Server Error	The server has encountered an error, check the server logfiles <code>catalina.log</code> and <code>stderr</code>

3.7.2 API Error Messages

The API server will send a message in the response body, should an error occur:



```
{
  "message": "Request method 'GET' not supported",
  "errorCode": "ITG-0A011",
  "stackTrace": "[org.springframework.web.servlet.mvc.m ...
  [...]
}
```

Error Code	Message
Client Errors	
ITG-0A001	Thin client with ID [id] not found.
ITG-0A002	No target directory with ID [id] found.
ITG-0A003	[command] is an invalid command.
ITG-0A004	Invalid request body
ITG-0A005	"Could not read JSON: [...]"
ITG-0A006	Directory with ID [id] is not empty. Remove content first.
ITG-0A007	No MAC defined.
ITG-0A008	[...] The statement was aborted because it would have caused a duplicate key value [...]
ITG-0A009	No firmware defined.
ITG-0A010	Number format error
ITG-0A011	Request method [method] not supported.
ITG-0A012	IDList is empty.
ITG-0A013	No valid property in body



ITG-0A014	MAC must be 12 characters long.
ITG-0A015	Firmware with ID [id] does not exist.
ITG-0A016	Parent with ID [id] does not exist.
ITG-0A017	Target directory is in Recycling Bin. In order to use it revert it first.
ITG-0A018	Thin client is in Recycling Bin. In order to use it, revert it first.
ITG-0A019	Invalid assignment receiver
ITG-0A020	No handler is mapped for this request.
ITG-0A021	Invalid assignment
ITG-0A022	Profile is in Recycling Bin.
ITG-0A023	Directory has Default Directory Rule
ITG-0A024	Invalid thin client name
Authentication Errors	
ITG-0B001	No authentication header found in request.
ITG-0B002	Invalid login credentials.
ITG-0B003	No session ID found.
ITG-0B004	No session found for requested session ID [id].
ITG-0B005	Access denied.
ITG-0B006	No valid IMI API version.
ITG-0B007	No user found in session with sessionID
Licensing Errors	
ITG-0C001	License required



ITG-0C002	Invalid IMI version
Server Errors	
ITG-FF001	Internal error
ITG-FF002	Database error
ITG-FF003	Cache error
ITG-FF004	Server not available
ITG-FF005	Object mapping error
ITG-FF006	Invalid object type



4 IMI How-Tos

- [Using Secure VNC via IGEL Management Interface \(IMI\)](#)(see page 193)

4.1 Using Secure VNC via IGEL Management Interface (IMI)

This document describes making SSL/TLS-secured VNC connections to *IGEL* Linux thin clients using the credentials provided by *IGEL Management Interface (IMI)* Version 2 .

4.1.1 Prerequisites

- *Universal Management Suite (UMS)* 5.02.100
- *IMI* maintenance subscription covering version 2
- *IGEL LX* thin clients with *IGEL Linux* version 5.03.190 or newer

Apart from that, you will have to implement a [custom VNC proxy](#)(see page 193).

4.1.2 Thin Client Configuration

In the thin clients' configuration, either locally, via UMS or in a profile, do the following:

1. Go to **System > Remote Access > Shadow**
2. Enable **Allow Remote Shadowing**
3. Optional: Enable **Prompt User to allow Remote Session**

In a number of countries, unannounced shadowing is prohibited by law. Do not disable this option if you are in one of these countries!

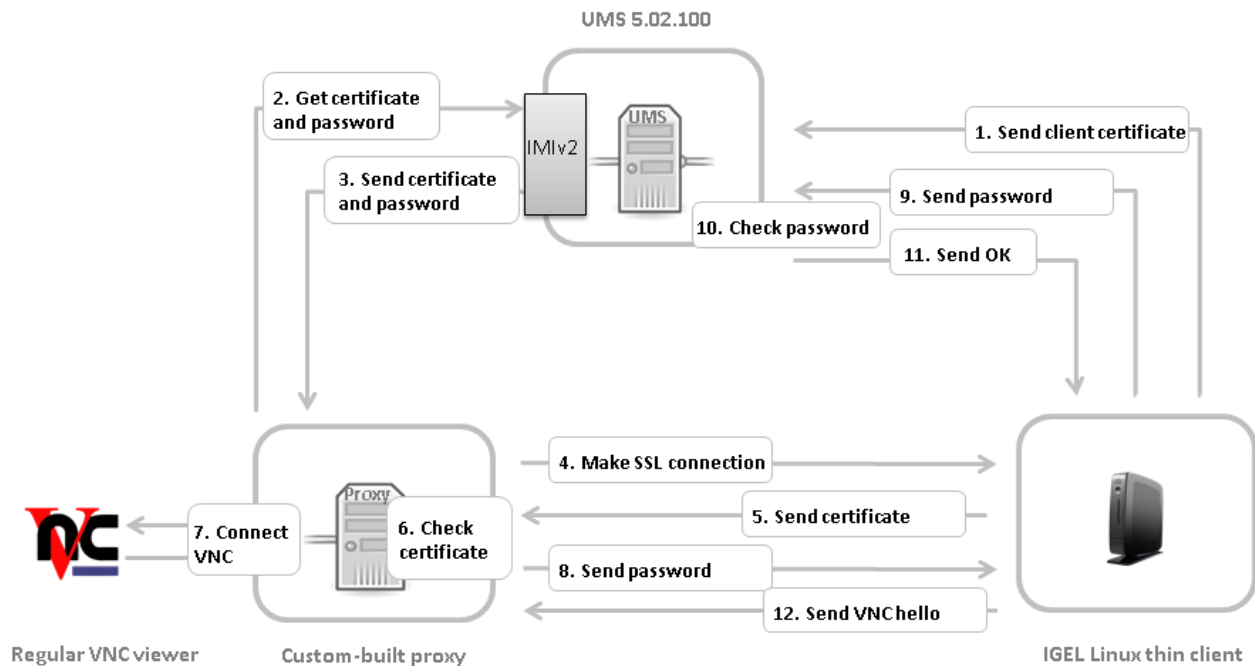
4. Optional: Enable **Allow Input from Remote**
5. Enable **Secure Mode**

Alternatively, you can globally activate Secure VNC for all thin clients under **UMS Administration > Remote Access > Secure VNC**.

6. Optional: Enable **Allow User to disconnect Remote Shadowing**

4.1.3 Secure VNC with IMIv2

This section describes the Secure VNC procedure, parts of which your custom VNC proxy has to implement.



1. The thin client sends its certificate to UMS on boot.

For a thin client newly registered to the UMS, either

- reboot the thin client once or
- use UMS Console to send **Settings UMS > TC** and send **Settings TC > UMS** once

2. Via the IMIv2 REST API, the custom proxy asks UMS for the credentials necessary for shadowing the thin client:
GET /umsapi/v2/thinclients/[thin client ID]?facets=shadow
3. The custom proxy receives the credentials:
 - the thin client certificate in Base64 encoding, or an empty string if UMS has not received a certificate
 - a one-time-password in the form of a Java UUID for logging in within the next 5 min.
4. The custom proxy initiates a SSL connection to the thin client's TCP port 5900.
5. The thin client sends its certificate.
6. The custom proxy checks the certificate against the one received from UMS and decides whether to accept it and proceed with the connection.
7. The custom proxy opens a TCP server socket for an external VNC viewer to connect to. The custom proxy must connect the network streams involved as follows:
 - write to the VNC viewer what it reads from the SSL connection with the thin client
 - write to the SSL connection with the thin client what it reads from the VNC viewer

Make the external VNC viewer connect to the custom proxy's server socket.

8. On accepting the connection from the external VNC viewer, the custom proxy writes the String PROCYCMD PW_[one-time password] to the SSL connection with the thin client.



9. The thin client sends the password to UMS.
10. UMS checks the password.
11. If the password is correct, UMS tells the thin client to proceed. Otherwise the connection will be closed.
12. The thin client sends a string like `RFB 003.008\n` as a VNC hello which initiates the VNC session with the external VNC viewer.