Miradore Online API

Specification Version 1.12

Contents

Change history 4
1. Introduction to Miradore Online API 6
2. URL and query string 7
2.1 Options 8
2.2 Select 9
2.3 Filters 11
2.4 OrderBy 13
3. Authentication and security 14
4. Available operations 15
4.1 Get 15
4.2 Create 16
4.3 Update 17
4.4 Remove 18
5. XML payload structure 20
5.1 Root element 20
5.2 Child elements 20
Appendix 1: Supported items and operations 22
Appendix 2: Supported attributes per item 23
AndroidSecurity 23
Client 23
ConfigProfile 23
ConfigProfileDeployment 24
Device 24
InvAndroidBattery 25
InvAndroidCamera 25
InvAndroidCpu 25
InvAndroidDisplay 25
InvAndroidSensor 26
InvAndroidWiFi 26
InvApplication 26
InvCertificate 26
InvDevice 27
InvIosProfile 27

© 2019 Miradore Ltd. All rights reserved.

iOSSecurity 27
InvOperatorNetwork 27
InvOS 28
InvSim 28
InvStorage 28
WindowsPhoneSecurity 29
Location 29
Organization 29
ReportedLocation 29
User 29
MobileSecurity 30
Tag 30
Enrollment 30

© 2019 Miradore Ltd. All rights reserved.

Change history

Version 1.12

Chapter	Description
Appendix 2	Added new attributes *LastReportedDays, LastReportedHours, LastReportedMinutes* for the *Device* item.
Appendix 2	Added a new attribute *AndroidID* for the *Device* item.
Appendix 2	Added new attributes *Created* and *Completed* for the *Enrollment* item.
Appendix 2	Added new attributes *Status* and *Source* for the *User* item.
Appendix 2	Added a new item *InvCertificate*.

Version 1.11

Chapter	Description
4.4	Added example 4.7 describing how to add a tag to a device
4.4	Added chapter 4.11 describing how to remove a tag from a device

Version 1.10

Chapter	Description
4.3	Added chapter 4.3 describing how to update child items of *List* type.
4.4	Added chapter 4.4 describing how to remove child items of *List* type.
Appendix 2	Added a new item *ConfigProfile.*
Appendix 2	Added a new item *ConfigProfileDeployment.*
Appendix 2	Added a new *ConfigProfile* attribute for the *Device* item.

Version 1.9

Chapter	Description
Appendix 2	Added new item *Enrollment.*

Version 1.8

Chapter	Description
Appendix 2	Added new attributes *IPAddress* and *MACAddress* for *Device* item.

Version 1.7

Chapter	Description
Appendix 2	Added new item *ReportedLocation.*

© 2019 Miradore Ltd. All rights reserved.

Version 1.6

Chapter	Description
Appendix 2	AndroidSecurity: Changed data type of DeviceAdministrationEnabled, PasscodeSufficient & Rooted from boolean to enumeration.
Appendix 2	WindowsPhoneSecurity: Changed data type of StorageCardDisabled from boolean to enumeration.

Version 1.5

Chapter	Description
Appendix 2	Device: Added Created, InvOS, InvSIM and Modified attributes
Appendix 2	InvDevice: Added DeviceName
Appendix 2	InvOperatorNetwork: CurrentCountryCode, HomeCountryCode, CurrentNetworkCode, HomeNetworkCode
Appendix 2	InvOS: Changed OSCategory to Platform
Appendix 2	Added new item InvSim
Appendix 2	Location: Added Created and Modified attributes
Appendix 2	Organization: Added Created and Modified attributes
Appendix 2	User: Added Created and Modified attributes

Version 1.4

Chapter	Description
Appendix 2	Added a new item Tag

Version 1.3

Chapter	Description
Appendix 2	Moved and renamed attributes OnlineStatus (old: Agent.Status), Platform (Agent.OS), LastReported (Agent.LastSeen) from Agent to Device.
Appendix 2	Renamed item Agent to Client.

Version 1.2

Chapter	Description
Appendix 2	Changes in Device item attributes. Removed attributes: InvWindowsPhoneSecurity, InvIosSecurity, InvAndroidSecurity Added attributes: Agent, Security
Appendix 2	Added a new item Agent
Appendix 2	Renamed item InvAndroidSecurity to AndroidSecurity, PasscodeSet and StorageEncryptionStatus attributes have been moved to MobileSecurity.
Appendix 2	Renamed item InvIosSecurity to iOSSecurity and PasscodePresent attribute has been moved to MobileSecurity.
Appendix 2	Renamed item InvWindowsPhoneSecurity to WindowsPhoneSecurity, DeviceEncryption and DeviceLock attributes have been moved to MobileSecurity.
Appendix 2	Added a new item MobileSecurity

Version 1.1
First published version

© 2019 Miradore Ltd. All rights reserved.

1. Introduction to Miradore Online API

Miradore Online API (Application Programming Interface) is an interface intended for integrating Miradore Online with external systems. This document describes the programming interface and includes lots of examples. The document is aimed for technical persons implementing the interface.

Miradore Online API is a REST based web service which can be used to programmatically create, read, update and remove items in Miradore Online. It is used over HTTPS with POST, GET, PUT and DELETE methods. API request are authenticated with authentication key in request's URL. Authentication keys are managed in Miradore Online web console. The API uses XML as its payload.

REST highly utilizes HTTP request URL in its operations. Chapter *2 URL and query string* describes how URLs are formed and what additional keywords can be used to modify queries. Chapter *3 Authentication and security* contains information about how web service calls can be authenticated. Chapter *4 Available operations* lists available operations and their usage. Chapter 5 XML payload structure describes the structure of input and output XML documents used by the web service. Appendix 1 contains a list of available configuration items and their attributes that can be accessed by the API.

© 2019 Miradore Ltd. All rights reserved.

2. URL and query string

Miradore Online API is used by sending HTTP requests to Miradore server. Depending on the operation, data is transferred in URL or the body of HTTP request. *Get* and *remove* operations transfer all required information in the HTTP request URL. *Create* and *update* operations require also some data in the HTTP request body. This chapter describes the structure of the web service HTTP request URL.

Base URL for API calls is:

https://<site>.online.miradore.com/API/<Item>

Table 1 lists the available operations and corresponding HTTP methods. For more information about operations, see 4 Available operations.

Table 1: HTTP methods used in different operations

Operation	Method	URL
Get	GET	/API/<Item> /API/<Item>/<ID>
Create	POST	/API/<Item>
Update	PUT	/API/<Item>/<ID>
Remove	DELETE	/API/<Item>/<ID>

The URL may contain an optional query string which contains specific settings to the query. Table 2 lists the available query string setting keywords. Query string settings are explained more comprehensively in the following chapters.

Table 2: Available query string setting (G=get, C=create, U=update and R=remove)

Setting	G	C	U	R	Description
auth	x	x	x	x	Authentication key generated in Miradore Online web console. This is required.
options	x	x	x	x	Can be used to define output and input formatting.
select	x	x	x		Defines what item attributes are selected to response.
filters	x				Defines item attribute based filters used to filter the items that query concerns.
orderby	x				Define the order in which the configuration items are listed within the response.

Query string is formatted just like any standard URL query string containing key-value pairs.

Example 2-1: Web service request with multiple keywords in query string

GET: */API/Device?auth=1_asi3tfoi2zD4Ht7&select=* InvDevice.*IMEI,* InvDevice.*Model,User.FirstName,User.LastName&filters=Location.Name eq Helsinki&options=rows=500,page=1*

This request contains *select*, *filters* and *options* keywords. *Options* defines two options: *rows* and *page.*

© 2019 Miradore Ltd. All rights reserved.

2.1 Options

Miradore Online API supports several options that can be used to modify query handling and response. Table 3 lists the available options. Detailed descriptions of the options can be found after the table.

Some of the options are on/off flags and some require value. The syntax for the options is the following:

/API/<Item>?options=flag,option=optionvalue

Table 3: Available query options

Option	*G*	*C*	*U*	*R*	*Description*	*Default*
dateformat	x	x	x		Defines the DateTime format string for input and output dates.	dd.MM.yyyy HH:mm:ss
page	x				Page number of the page which will be returned.	1
rows	x				Number of rows (root level items) on one page.	100
usenamespace	x	x	x		If set, adds an XML namespace declaration to response documents.	Not set

Dateformat

Dateformat option defines the representation of incoming and outgoing date-time values. With this option the API can be used with a date format familiar for caller end. *Dateformat* supports *.Net* date format strings. Refer to Microsoft's documentation for more information about available configurations. For example, information about custom date and time format strings can be found from http://msdn.microsoft.com/en-us/library/8kb3ddd4.aspx.

- **Default value:** dd.MM.yyyy HH:mm:ss
- **Usage:** options=dateformat=<.Net DateTime format string>

Example 2-2: Output when option dateformat=MM-dd-yyyy is set

```
<Content>
    <Items count="1">
        <Device>
            <WarrantyEndDate>01-20-2014</WarrantyEndDate>
        </Device>
    </Items>
</Content>
```

Paging

Paging can be used to limit the number of items returned by the query. If the query returns lots of items, it might be useful to improve query performance by processing data in smaller pieces. Paging can be set with the following options:

- **Rows:** Defines the number of items returned per page (default: 100)
- **Page:** Page number (default: 1)

© 2019 Miradore Ltd. All rights reserved.

Example 2-3: Usage of paging with rows and page options

For example, if your query returns 550 items, you can request the data in three consecutive queries:

- **Query #1:** options=rows=250,page=1
- **Output:** Items from 1 to 250

- **Query #2:** options=rows=250,page=2
- **Output:** Items from 251 to 500

- **Query #3:** options=rows=250,page=3
- **Output:** Items from 501 to 550

Usenamespace

This option can be used to add default namespace declaration to the response documents. See *5.1 Root element* for more information.

- **Default value: -**
- **Usage:** options= usenamespace

Example 2-4: Usage of usenamespace option

- **Usage:** GET: *API/Device/1835?Options=usenamespace&select*=InvDevice.*IMEI*
- **Example output:**

```
<Content xmlns="http://www.online.miradore.com/xmlns/api/1.0">
    <Items count="1">
        <Device>
            <InvDevice>
                <IMEI>012804006425674</IMEI>
            </InvDevice>
        </Device>
    </Items>
</Content>
```

2.2 Select

Select query setting is used to define which item attributes are included in the response.

Each exportable item has a predefined list of selected attributes. These default attributes are returned if nothing else is selected. Best practice is always to define explicitly the attributes required by the caller end.

If select is used with *create* or *update* operations, created or updated item is returned. Without *select* those operations return empty response.

The syntax for the expression is the following:

/API/<Item>?select=Attribute1,Attriburte2,ChildItem1.ChildAttribute1,ChildItem1.ChildAttribute2

Following examples explain the usage of the *select* setting.

Example 2-5: Simple select example

Select only WarrantyEndDate, User.LastName and User.FirstName attributes.

- **Usage**: GET: *API/Device*?auth=1_asdf&select=WarrantyEndDate,User.LastName,User.FirstName

© 2019 Miradore Ltd. All rights reserved.

- **Example output:**

```
<Content>
    <Items count="1">
        <Device>
            <WarrantyEndDate>21.08.2015 03.00.00</WarrantyEndDate>
            <User>
                <Lastname>John</Lastname>
                <Firstname>Straw</Firstname>
            </User>
        </Device>
    </Items>
</Content>
```

Example 2-6: Select with filtering

The *select* setting can be used in conjunction with *filters* setting (see *2.3 Filters*). Excluding an attribute with the *select* does not prevent filtering with the attribute.

Select user's first and last names and warranty end dates of devices which inventoried model name contains word *iPhone*

- **Usage:** GET: *API/Device*?auth=1_asdf&select=User.FirstName,User.LastName,WarrantyEndDate &filters=InvDevice.Model contains iPhone
- **Example output:**

```
<Content>
    <Items count="1">
        <Device>
            <User>
                <Firstname>Jim</Firstname>
                <Lastname>Straw</Lastname>
            </User>
            <WarrantyEndDate>24.08.2014 03.00.00</WarrantyEndDate>
        </Device>
    </Items>
</Content>
```

Example 2-7: "Select all" statement

By default, Miradore Online API returns only default attributes for an item. *Select all* statement can be used to select all attributes, except inventoried attributes, from an item. '*' must be defined explicitly for each child item if all attributes are wanted to select. I.e. '*' on the root level applies only attributes of the requested item and for child items, only the default set of attributes is selected.

Select all attributes of *Device* (queried item), all attributes of *User* child item and *Model* attribute of *InvDevice* child item. As you can see from the output, only default attributes for *Location* and *Organization* child items are selected because query doesn't contain explicitly defined select list for them.

- **Usage:** GET: *API/Device*?auth=1_asdf&select=*,User.*,InvDevice.Model

© 2019 Miradore Ltd. All rights reserved.

- **Example output:**


```
<Content>
    <Items count="1">
        <Device>
            <ID>1008</ID>
            <User>
                <ID>1024</ID>
                <Name>Jim Straw</Name>
                <Email>jim.straw@trestacom.com</Email>
                <Firstname>Jim</Firstname>
                <Lastname>Straw</Lastname>
                <PhoneNumber>+358 50 1234 567</PhoneNumber>
            </User>
            <Location>
                <ID>2</ID>
                <Parent>
                    <ID>1</ID>
                </Parent>
            </Location>
            <Organization>
                <ID>4</ID>
                <Parent>
                    <ID>1</ID>
                </Parent>
            </Organization>
            <Status>AutoGenerated</Status>
            <WarrantyEndDate>24.08.2014 03.00.00</WarrantyEndDate>
            <InvDevice>
                <Model>Apple iPhone5,2</Model>
            </InvDevice>
        </Device>
    </Items>
</Content>
```


2.3 Filters

API queries can be filtered by adding *filters* parameter in request URL's query string. *Filters* specify conditions that must be met by a resource for it to be returned in the response.

Filter must contain at least item's attribute, comparison operator and value. *Table 4* lists supported comparison operators and shows examples of their usage. The syntax for the single value filter expression is as follows:

/Item?filters=<Item.Attribute1> <comparison_operator> <value>

Table 4: Supported comparison operators

Operator	*Description*
eq	Equal to
ne	Not equal to
gt	Greater than
ge	Greater than or equal to
lt	Less than
le	Less than or equal
contains	Value contains defined string
notcontains	Value doesn't contain defined string
startswith	Value starts with defined string
endswith	Value ends with defined string
isempty	Value is empty
notisempty	Value is not empty

© 2019 Miradore Ltd. All rights reserved.

Miradore Online API supports also combining of multiple attribute comparisons with logical operators. Table 5 lists supported logical operators and shows examples of their usage. The syntax for filtering expression with logical operators is as follows:

/API/<Item>?filters=< Attribute1> <comparison_operator1> <value1> <logical_operator> <Attribute2> <comparison_operator2> <value2>

Table 5: Supported logical operators

Operator	*Example*
and	PurchaseDate gt '01.01.2013 00.00.00' **and** InvDevice.Model contains iPhone
or	InvDevice.Model contains iPhone **or** InvDevice.Model contains Galaxy

Precedence of *and* and *or* operators is equal and expressions are evaluated from left to right. Brackets can be used to change the order of evaluation of expressions.

Following examples demonstrates the usage of *filters* setting.

Example 2-8: Filtering by one attribute

Get devices which are purchased since January 1, 2013

- **Usage:** GET /API/Device?auth=1_asdf&filters=PurchaseDate gt ’01.01.2013 00.00.00’

Example 2-9: Filtering with brackets

Brackets can be used to change the precedence of AND and OR logical operators.

Get devices which manufacturer is *Nokia* or *Apple* and which are purchased since *January 1, 2013* or which warranty ends before *January 1, 2015*.

- **Usage:** GET /API/Device?auth=1_asdf&filters=(InvDevice.Manufacturer eq Nokia or InvDevice.Manufacturer eq Apple) and (PurchaseDate gt ‘01.01.2013 00.00.00’ or WarrantyEndDate lt ‘01.01.2015 00.00.00’)

Example 2-10: Filtering by values with empty spaces

Filters and filter operators are separated by empty spaces, and therefore, single or double quotation marks are required if the filtered value contains empty spaces.

- **Usage:** GET /API/Device?auth=1_asdf&filters=Organisation.Name eq 'Research and Development'

If filter string contains single quotation marks double quotation marks must be used around the value and vice versa.

- **Usage:** GET /API/Device?auth=1_asdf&filters=Organisation.Name eq "Where's My Droid"

© 2019 Miradore Ltd. All rights reserved.

2.4 OrderBy

Order by query setting can be used to sort the results by one or multiple attributes delimited by commas. Sorting can be done based on any attribute except attributes that can contain multiple values like device's InvApplication. Attribute used for sorting must be also defined in query's select list. Sorting can be done in either ascending (*asc*) which is the default or descending (*desc*) order.

Example 2-11: Orderby

Order by device purchase date in descending order:

- **Usage:** GET /API/Device?auth=1_asdf&orderby=PurchaseDate desc&select=PurchaseDate

Order by user's last name and first name:

- **Usage:** GET /API/Device?auth=1_asdf&oderby=User.LastName,User.FirstName&select=User.LastName,User.FirstName

© 2019 Miradore Ltd. All rights reserved.

3. Authentication and security

Miradore Online API uses authentication key in request's URL query string for authentication. Authentication key is random string that can be generated from Miradore Online web console's *Infrastructure diagram* after the API feature has been activated. Keys can also be deleted from the same place when they are not needed anymore. It's recommended to generate separate authentication keys for each use case. This way administrator can easily manage generated keys.

Example 3-1: Auth query string parameter

Auth query string parameter with authentication key as value must be defined in each API request

- **Usage:** GET /API/Device?auth=1_AaDf234sdf8!4

© 2019 Miradore Ltd. All rights reserved.

4. Available operations

This chapter explains operations available through Miradore Online API. It supports four different operations: *get, create, update* and *remove.* Each operation is explained more comprehensively below in operation specific sections.

4.1 Get

Get operation is used to get list of configuration items with selected attributes from Miradore Online.

Request

Get can be used to get single item with item ID. In this case item ID is defined in the path partition of URL. Another option is to define filters in query string and query returns all items which to filter matches. Refer *2 URL and query string* for information about query options.

Example 4-1: Get operation with item id and select list

GET: */API/Device/1?auth*=1_AaDf234sdf8!4*&select=User.LastName,User.FirstName,* InvDevice.*Model*

RESPONSE:

```
<Content>
    <Items count="1">
        <Device>
            <User>
                <Lastname>Straw</Lastname>
                <Firstname>Jim</Firstname>
            </User>
            <InvDevice>
                <Model>LGE Nexus 4</Model>
            </InvDevice>
        </Device>
    </Items>
</Content>
```

Query returns device which *ID* is 1 which is defined in URL. Query string contains also *auth* parameter which is used for authentication and *select* setting which contains list of selected attributes.

Example 4-2: Get operation with filters

GET: */API/Device/1?auth*=1_AaDf234sdf8!4*&select=User.LastName,User.FirstName,* InvDevice.*Model&filters=Location.Name eq Helsinki*

RESPONSE:

```
<Content>
    <Items count="2">
        <Device>
            <User>
                <Lastname>Straw</Lastname>
                <Firstname>Jim</Firstname>
            </User>
            <InvDevice>
                <Model>samsung GT-I9300</Model>
            </InvDevice>
        </Device>
        <Device>
            <User>
                <Lastname>Straw</Lastname>
                <Firstname>John</Firstname>
            </User>
            <InvDevice>
                <Model>LGE Nexus 4</Model>
            </InvDevice>
        </Device>
    </Items>
</Content>
```

© 2019 Miradore Ltd. All rights reserved.

Query returns devices which location name is *Helsinki*. This is defined by *filters* setting in query string. Query string contains also *auth* parameter which is used for authentication and *select* setting which contains list of selected attributes.

Response

Get operation returns HTTP status codes defined in *Table 6*. Response contains found items in XML document. See examples from *Request* section above.

Table 6: Get specific http status codes

Status code	Description
200	Get operation executed successfully
400	Bad request, check response's Error element
401	Authentication failed, *auth* query string parameter is missing or incorrect
500	Internal server error has been occurred

4.2 Create

Create operation is used to create new items to Miradore Online.

Only one item can be created with a single call. This makes error handling and identification of created items easier.

If item already exists i.e. some of item's attributes that should be unique is already reserved by some existing item, error is returned.

Request

Create uses HTTP POST method and created item is send in requests body.

Created item must be defined in request XML document within Content/*Items* element. See following examples and *5 XML payload structure* for more information about structure of XML document.

Query string setting *select* can be used to define which created item's attributes are returned. Otherwise, empty HTTP response is returned if item is created succesfully.

Example 4-3: Create user

POST: /API/User*?auth*=1_AaDf234sdf8!4&select=ID,Name

BODY:

```
<Content>
    <Items>
        <User>
            <Email>jim.straw@trestacom.com</Email>
            <Firstname>Jim</Firstname>
            <Lastname>Straw</Lastname>
            <PhoneNumber>123456789</PhoneNumber>
        </User>
    </Items>
</Content>
```

Response

Create operation returns HTTP status codes defined in *Table 7*. If *select* clause is added to query, response's body contains created item with selected attributes. This XML document is formatted same way as responses of *get* operation.

© 2019 Miradore Ltd. All rights reserved.

Table 7: Create specific HTTP status codes

Status code	Description
200	All items are created successfully
202	Item has been created but there is some warnings, check response's Warnings element
400	Bad request, check response's /Response/Error element
500	Internal server error has been occurred

Example 4-4: Response of create user request

Contents of the previous example's create user request:

```
<Content>
     <Items count="1">
          <User>
               <ID>1055</ID>
               <Name>Straw Jim</Name>
          </User>
     </Items>
</Content>
```

4.3 Update

Update operation is used to update one or more attributes of a specified item. The item must be specified with the item *name* and *id*.

Request

The *Update* operation is used with the *HTTP PUT* method by sending a request XML document to the URL which identifies the item to be updated. The *R*equest XML document must contain the updated values of the item's attributes. The attributes which are not defined within the request are left unmodified.

An attribute can be reset to empty value by adding an empty attribute element to the request.

The query string setting *select* can be used to define which updated item's attributes are returned.

Example 4-5: Update user which ID is 1055. Set middle name to M

PUT: /API/User/1055*?auth*=1_AaDf234sdf8!4

BODY:

```
<Content>
    <Items>
         <User>
              <Middle>M</Middle>
         </User>
    </Items>
</Content>
```

The attributes whose type is *List,* like *Device's ConfigProfileDeployment*, are updated with a special method. A new child item is always added to the list when the update command is called. Items can be removed from the list with the *Remove* command.

The syntax for adding items to the child item list is:

PUT: /API/<Item>/<ItemID>/<ChildItem>/<ChildItemID>

© 2019 Miradore Ltd. All rights reserved.

Where:

<Item> = Target item name

<ItemID> = Target item's ID attribute value which uniquely identifies it

<ChildItem> = Child item attribute name

<ChildItemID> = Child item's ID attribute value which uniquely identifies child item

In this case, the *Update* command doesn't require anything in the HTTP request's body.

Example 4-6: Deploy configuration profile to a device

PUT: /API/Device/105/ConfigProfile/5*?auth*=1_AaDf234sdf8!4

This query will deploy a *ConfigProfile* with ID = 5 to the Device with ID = 105.

Example 4-7: Add tag to a device

PUT: /API/Device/105/Tag/BYOD Android device*?auth*=1_AaDf234sdf8!4

This query will add a *Tag* with Name = BYOD Android device to the Device with ID = 105.

Response

The *Update* operation's HTTP status codes are defined in *Table 8*. If the *select* clause is added to the query, the response's body contains the updated item with the selected attributes. This XML document is formatted similarly as the responses of the *get* operation.

Table 8: Update specific HTTP status codes

Status code	*Description*
200	Item updated successfully
400	Bad request, check response's /Response/Error element
500	Internal server error has been occurred

4.4 Remove

Remove operation is used to remove an item from Miradore Online. Only one item can be removed with a single call and the removed item must be identified with the item *ID*.

Request

The *Remove* operation is used with the *HTTP DELETE* method by sending a request to the URL which identifies the item to be removed.

Example 4-8: Remove asset which AssetID is 123

DELETE /API/User/1055*?auth*=1_AaDf234sdf8!4

The child item attributes whose type is *List*, like Device's *ConfigProfileDeployment*, are removed with a special method. The remove command is always targeted to parent item and child item is defined in the request's URL. When removing a single child item, the child item ID must also be defined within the request URL. The syntax is described below:

DELETE: /API/<Item>/<ItemID>/<ChildItem>/<ChildItemID>

© 2019 Miradore Ltd. All rights reserved.

Where:

<Item>	= Target item name
<ItemID>	= Target item's ID attribute value which uniquely identifies it
<ChildItem>	= Child item attribute name
<ChildItemID>	= Child item's ID attribute value which uniquely identifies child item

Example 4-9: Remove a ConfigProfile from target Device

DELETE /API/Device/105/ConfigProfile/5

This query will remove the *ConfigProfile* with ID = 5 from the *Device* with ID = 105.

All *List* child items are removed when child item ID is not defined in the request URL. In this case syntax looks this:
DELETE: /API/<Item>/<ItemID>/<ChildItem>

Example 4-10: Remove all ConfigProfiles from target Device

DELETE /API/Device/105/ConfigProfile

This query removes all *ConfigProfiles* from the device with ID = 105.

Example 4-11: Remove a Tag from target Device

DELETE /API/Device/105/Tag/BYOD Android device

This query will remove the *Tag* with Name = BYOD Android device from the *Device* with ID = 105.

Response

The *Remove* operation returns an empty HTTP response with status code 200 if the removal succeeds. Otherwise, it returns a response XML document with *Error* element.

Table 9: Remove specific HTTP status codes

Status code	*Description*
200	Removed successfully, response has no content
400	Bad request, check response's /Response/Error element
500	Internal server error has been occurred

© 2019 Miradore Ltd. All rights reserved.

5. XML payload structure

This chapter describes the structure of web service XML payload.

5.1 Root element

Miradore Online API always returns response XML document which contains *Content* root element.

Table 10: List of attributes in Content element

Attribute	Description
xmlns	Contains XML namespace declaration *http://www.online.miradore.com/xmlns/api/1.0.*

xmlns

This contains namespace declaration for web service responses if usage of namespace is enabled with Usenamespace option. Web service uses default namespace *http://www.miradore.com/xmlns/ws/1.0*.

5.2 Child elements

Items

This element contains the actual data of the query i.e. a list of items of certain type. Response document's *Items* element can contain zero or more items as child elements. Refer *Appendix 1: Supported items and operations* to see the list of supported items.

API adds attributes defined in *Table 11* to response document's *Items* element.

Table 11: Attributes of Items element

Attribute	Description
count	Total count of items matched to the filters of the query.

Items element's child elements are named with queried item's name, like *Device* element in *Example 5-1*. All items have attributes which are represented as child elements under configuration item element, like *Location* and *User* elements in the example. Refer *Appendix 2: Supported attributes per item* to see the list of available attributes.

Attributes can also be items (so called child items). Typically these child item elements are named with child item name like Device's *Location* but names may also differ.

© 2019 Miradore Ltd. All rights reserved.

Example 5-1: Items element in response document

```
<Content count="24">
    <Items>
    <Device>
         <User>
              <ID>1</ID>
              <Name>Jim Straw</Name>
              <Email>jim.straw@trestacom.com</Email>
              <Firstname>Jim</Firstname>
              <Lastname>Straw</Lastname>
              <PhoneNumber>+358123456798</PhoneNumber>
         </User>
         <Location>
              <ID>3</ID>
              <Name>Helsinki</Name>
              <FullName>Finland > Helsinki</FullName>
              <Parent>
                   <ID>1</ID>
              </Parent>
         </Location>
    </Device>
</Items>
```

Error

This element contains description of occurred error. It has following child elements *Code (*HTTP status code)*, Description* (human readable error description). If error occurs, processing of web service call is aborted immediately. Following example contains *Error* element.

Example 5-2: Response which contains error

```
<Content>
    <Error>
         <Description>Entity 'Asset' does not have property 'Usr'.</Description>
         <Code>400</Code>
    </Error>
</Content>
```

There may also be higher level errors like API call's URL path is not defined correctly or unsupported HTTP method or content type is used. In this case server returns more generic error message which status code and contents depends on the type of error.

© 2019 Miradore Ltd. All rights reserved.

Appendix 1: Supported items and operations

Following table lists supported items and operations which are available for them.

Item	*Get*	*Create*	*Update*	*Remove*
Device	x			
Location	x			
Organization	x			
User	x	X	X	X

© 2019 Miradore Ltd. All rights reserved.

Appendix 2: Supported attributes per item

Following tables list all available items and their attributes.

Data type column contains the type of attribute. This may contain *.Net* data type or some of the abbreviations listed in the following table. *Child item* column contains the item name of the child item. You can find details about the child item with this name from the documentation. Typically *Child item* name is same as *Attribute* name but it may also differ. *Values* column contains allowed attribute values if attribute can have only some of predefined values.

Abbreviation	*Meaning*	*Description*
C	Child item	Child items are another configuration items which are linked to parent item. *Child item* column defines the actual type of child item.
L	List	List typed attributes can contain multiple instances of item defined by *Child item* column.
E	Enumeration	Enumerations can contain some predefined value. Allowed values are listed in *Values* column.

Items which supports other than *get* operation has *G* (get), *C* (create), *U* (update) and *R* (remove) columns which tell is attribute available for those operations.

AndroidSecurity

Attribute	*Data type*	*Child item*	*Values*
DeviceAdministrationEnabled	E		Unknown, Disabled, Enabled
InventoryTime	DateTime		
PasscodeSufficient	E		Unknown, No, Yes
Rooted	E		Unknown, NotRooted, Rooted
SAFEStatus	E		Disabled, Enabled, NotSupported, Unknown
StorageEncryptionRequired	E		No, NotSupported, Unknown, Yes

Client

Attribute	*Data type*	*Child item*	*Values*
ID	Int32		
Version	String		

ConfigProfile

Attribute	*Data type*	*Child item*	*Values*
ID	Int32		
ConfigurationType	String		
Description	String		
Name	String		
OSCategory	E		Android, iOS, WindowsPhone

© 2019 Miradore Ltd. All rights reserved.

ConfigProfileDeployment

Attribute	Data type	Child item	Values
ID	Int32		
ConfigProfile	C	ConfigProfile	
DeploymentTime	DateTime		
DeploymentTrigger	E		Administrator, BusinessEnforcement

Device

Attribute	Data type	Child item	Values	G	C	U	R
Client	C	Client		x			
Created	DateTime			x			
ConfigProfileDeployment	L	ConfigProfileDeployment		x		x	x
ID	Int32			x			
InvAndroidBattery	C	InvAndroidBattery		x			
InvAndroidCamera	L	InvAndroidCamera		x			
InvAndroidCpu	C	InvAndroidCpu		x			
InvAndroidDisplay	L	InvAndroidDisplay		x			
InvAndroidSensor	L	InvAndroidSensor		x			
InvAndroidWiFi	L	InvAndroidWiFi		x			
InvApplication	L	InvApplication		x			
InvDevice	C	InvDevice		x			
InvIosProfile	L	InvIosProfile		x			
InvCertificate	L	InvCertificate		x			
InvOperatorNetwork	C	InvOperatorNetwork		x			
InvOS	C	InvOS		x			
InvSIM	C	InvSIM		x			
InvStorage	L	InvStorage		x			
LastReported	DateTime			x			
LastReportedDays	Int32			x			
LastReportedHours	Int32			x			
LastReportedMinutes	Int32			x			
Location	C	Location		x			
Modified	DateTime			x			
OnlineStatus	E		Active, Inactive, Unavailable, Unknown	x			
Organization	C	Organization		x			
Platform	E		Android, iOS, Unknown, WindowsPhone, WindowsDesktop, macOS	x			
PurchaseDate	DateTime			x			

© 2019 Miradore Ltd. All rights reserved.

ReportedLocation	C	ReportedLocation		x
Security	C	MobileSecurity		x
Status	E		Active, AutoGenerated, Deleted, New	x
Source	E		MiradoreOnline	x
Tag	L	Tag		x
User	C	User		x
WarrantyEndDate	DateTime			x
IPAddress	String			x
MACAddress	String			x
AndroidID	String			x
Enrollment	C	Enrollment		x

InvAndroidBattery

Attribute	*Data type*	*Child item*	*Values*
ChargeLevel	Decimal		
Health	E		Cold, Dead, Good, Overheat, OverVoltage, Unknown, Unspecified
InventoryTime	DateTime		
Technology	String		
Temperature	Decimal		
Voltage	Decimal		

InvAndroidCamera

Attribute	*Data type*	*Child item*	*Values*
FocalLength	Decimal		
HasFlash	Boolean		
InventoryTime	DateTime		
MegaPixels	Decimal		

InvAndroidCpu

Attribute	*Data type*	*Child item*	*Values*
Architecture	String		
CoreCount	Int32		
InstructionSet	String		
InventoryTime	DateTime		
MaxFrequency	Int32		

InvAndroidDisplay

Attribute	*Data type*	*Child item*	*Values*
Density	String		
InventoryTime	DateTime		
PhysicalSize	Decimal		
RefreshRate	Decimal		
Resolution	String		

© 2019 Miradore Ltd. All rights reserved.

InvAndroidSensor

Attribute	Data type	Child item	Values
InventoryTime	DateTime		
Name	String		
PowerUsage	Decimal		
Type	E		Accelerometer, AmbientTemperature, Gravity, Gyroscope, Light, LinearAcceleration, MagneticField, Pressure, Proximity, RelativeHumidity, RotationVector, Temperature, Unknown
Vendor	String		

InvAndroidWiFi

Attribute	Data type	Child item	Values
InventoryTime	DateTime		
SSID	String		
Status	E		Current, Disabled, Enabled, Unknown

InvApplication

Attribute	Data type	Child item	Values
Identifier	String		
InventoryTime	DateTime		
IsLauncher	Boolean		
Name	String		
OSCategory	E		Android, iOS, Unknown, WindowsPhone
Size	Int64		
Version	String		

InvCertificate

Attribute	Data type	Child item	Values
CommonName	String		
IsIdentity	Boolean		
IssuedBy	String		
Subject	String		
Thumbprint	String		
ValidFrom	DateTime		
ValidTo	DateTime		

© 2019 Miradore Ltd. All rights reserved.

InvDevice

Attribute	Data type	Child item	Values
BluetoothMAC	String		
DeviceName	String		
DeviceType	String		
DoNotDisturbActive	Boolean		
EASIdentifier	String		
IMEI	String		
InventoryTime	DateTime		
Manufacturer	String		
Model	String		
Motherboard	String		
ProductName	String		
SerialNumber	String		
SoftwareVersion	String		
StoreAccountActive	Boolean		
UDID	String		
WiFiMAC	String		

InvIosProfile

Attribute	Data type	Child item	Values
Description	String		
Identifier	String		
InventoryTime	DateTime		
Name	String		
RemovalAllowed	Boolean		

iOSSecurity

Attribute	Data type	Child item	Values
ActivationLock	Boolean		
DeviceLocatorService	Boolean		
HardwareEncryption	E		BlockAndFileLevelEncryption, BlockLevelEncryption, FileLevelEncryption, Unknown
InventoryTime	DateTime		
PasscodeCompliant	Boolean		
PasscodeCompliantWithProfiles	Boolean		
Supervised	Boolean		

InvOperatorNetwork

Attribute	Data type	Child item	Values
CellID	Int32		
CurrentCountry	String		
CurrentCountryCode	Int32		

© 2019 Miradore Ltd. All rights reserved.

CurrentNetwork	String		
CurrentNetworkCode	String		
CurrentRoamingState	Boolean		
DataRoaming	Boolean		
HomeCountry	String		
HomeCountryCode	Int32		
HomeNetwork	String		
HomeNetworkCode	String		
HotspotEnabled	Boolean		
InventoryTime	DateTime		
LocationAreaCode	Int32		
NetworkType	E		CDMA, EDGE, EHRPD, EVDO_0, EVDO_A, EVDO_B, GPRS, HSDPA, HSPA, HSPAP, HSUPA, IDEN, LTE, ONExRTT, UMTS
OperatorName	String		

InvOS

Attribute	*Data type*	*Child item*	*Values*
Build	String		
InventoryTime	DateTime		
Language	String		
Platform	E		Android, iOS, Unknown, WindowsPhone, WindowsDesktop, macOS
Version	String		

InvSim

Attribute	*Data type*	*Child item*	*Values*
ICCID	String		
IMSI	String		
InventoryTime	DateTime		
PhoneNumber	String		

InvStorage

Attribute	*Data type*	*Child item*	*Values*
FreeSpace	Int64		
InventoryTime	DateTime		
TotalSpace	Int64		
Type	E		Internal, Optical, Remote, Removable, Undetermined
Volume	String		

© 2019 Miradore Ltd. All rights reserved.

WindowsPhoneSecurity

Attribute	Data type	Child item	Values
InventoryTime	DateTime		
StorageCardDisabled	E		Unknown, No, Yes

Location

Attribute	Data type	Child item	Values
Created	DateTime		
FullName	String		
ID	Int32		
Modified	DateTime		
Name	String		
Parent	C	Location	

Organization

Attribute	Data type	Child item	Values
Created	DateTime		
FullName	String		
ID	Int32		
Modified	DateTime		
Name	String		
Parent	C	Organization	

ReportedLocation

Attribute	Data type	Child item	Values
City	String		
Country	String		
FixTime	DateTime		
InventoryTime	DateTime		
Latitude	String		
Longitude	String		
StreetAddress	String		
ZipCode	String		

User

Attribute	Data type	Child item	Values	G	C	U
Created	DateTime					
Email	String			x	x	x
Firstname	String			x	x	x
ID	Int32			x		
Lastname	String			x	x	x
Middle	String			x	x	x
Modified	DateTime					

© 2019 Miradore Ltd. All rights reserved.

Name	String		x		
PhoneNumber	String		x	x	x
Status	E	New, Active, Retired, System	x		
Source	E	Unknown, GUI, CSV, API, AD	x		

MobileSecurity

Attribute	*Data type*	*Child item*	*Values*
PasscodeSet	E		No, Unknown, Yes
EncryptionStatus	E		Activating, Disabled, Enabled, NotSupported, Unknown
Android	C	AndroidSecurity	
iOS	C	iOSSecurity	
WindowsPhone	C	WindowsPhoneSecurity	

Tag

Attribute	*Data type*	*Child item*	*Values*
Name	String		

Enrollment

Attribute	*Data type*	*Child item*	*Values*
ID	Int32		
User	C		
Created	DateTime		
Completed	DateTime		
Type	E		SMSAndEmail, SMS, Email, MasterKey, ShowCredentials, DEP
Sender	String		

© 2019 Miradore Ltd. All rights reserved.