Depsim User Guide

Description

depsim provides server emulation of the APIs for the Device Enrollment Program documented in the Mobile Device Management Protocol Reference.

Synopsis

depsim <**command**> [[<**option**> [<**argument**> …]] …]

Commands

start [**-port** **port**] [**-output** **path**] [**configpath** …]

Start an instance of depsim in server mode. By default, no devices are registered and server activity is logged in JSON format to stdout. The default or specified configuration is output at the start, including the OAuth credentials required for authentication. The process listens on port 8080. Use the option -**port** to specify an alternate port. Use the option -**output** to redirect logs to the specified file location. Specify one or more optional configuration files to override the default configuration.

stop [**-port** **port**] [**-host** **address**]

Stop an instance of depsim in server mode. By default, the command will attempt to stop a server instance running on the local host at port 8080. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host.

configure [**-port** **port**] [**-host** **address**] **configpath** …

Load one or more configuration files into a server instance. One or more configuration files, separated by spaces, must be specified following the command. Values in the specified configurations will immediately override settings in the running server instance. Overriding the settings of a repeating operation, such as device record creation, will cause it to stop and commence its newly specified behavior. By default, this command will attempt to connect to a server instance running on the local host at port 8080. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host.

reset [**-port** port] [**-host** address] [**-preserve_sessions**]

Reset a server instance to the originally specified configuration. By default, this command will attempt to connect to a server instance running on the local host at port 8080. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host. Use the option -**preserve_sessions** to prevent clearing active session tokens.

log [**-port** port] [**-host** address] [**-output** path]

Output the current stream of server log entries to stdout. By default, this command will attempt to connect to a server instance running on the localhost at port 8080. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host. If -**output** is specified the contents of the log will be written to the file at the given path.

createdevice {[**-port** port] [**-host** address] | **-output** [**path**]} [**count** number] [**type** …]

Create a new device record. By default, this command will attempt to connect to a server instance running on the local host at port 8080 and add a randomly generated device record. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host. Use the option -**count** to specify the number of devices to create. Specify one or more type(s) of device(s) to override the default device types. The valid values are "IPAD", "IPHONE", "IPOD", "MAC". Use the option -**output** to write the created device(s) to stdout or a specified file on the local host instead of a running server instance. If -**output** is specified, -**port** and -**host** are ignored.

deletedevices [**-port** port] [**-host** address] {**serial_number … | -input path … | -all**}

Delete existing device records. One or more device serial numbers separated by spaces must be specified, or the option -**input** followed by one or more file paths for JSON formatted files containing device serial numbers must be specified or the option -**all** must be specified. By default, this command will attempt to connect to a server instance running on the local host at port 8080. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host.

dumpdevices [**-port** port] [**-host** address] [**-output** path]

Output the current list of devices loaded in a server instance. The device records are output in JSON format as an array of dictionaries. There is one dictionary entry per device. By default, this command will attempt to connect to a server instance running on the local host at port 8080 and write the device records to stdout. Use the option -**port** to specify an alternate port. Use the option -**host** to specify an alternate host. Use the option -**output** to write the device records to a specified file on the local host.

dumpprofile [-port port] [-host address] [-output path] {-serial_number … | -profile_uuid …}

Output the current profile for the device on the server. The profile record is output in JSON format and there is one profile entry per device. The device record is specified by the serial number or the profile uuid. By default, this command will attempt to connect to a server instance running on the local host at port 8080 and write the device records to stdout. Use the option -port to specify an alternate port. Use the option -host to specify an alternate host. Use the option -output to write the device records to a specified file on the local host.

license

Display the end user software license agreement.

acknowledgements

Display third-party acknowledgments.

help

Display the complete documentation.

Options

-port <number>

(All commands) Specify the network port.

-host <hostname>

(All commands) Specify the hostname.

-count <number>

(createdevice command only) Specify the number of randomly generated device records to create.

-output [<file>]

(start, log, stoken, createdevice and dumpdevices commands only) Specify the output destination for the command. An optional file path on the local host can be specified. If a file path is not specified, the output is sent to stdout.

-input <files>

(deletedevices command only) Specify one or more file paths on the local host whose file contents contain newline separated serial numbers of devices to delete from the server.

-all

(deletedevices command only) Delete all the device records in the server.

-preserve_sessions

(reset command only) Don't clear currently active session tokens when reseting the server to the originally specified configuration.

Examples

To start the server and have it listen on port 9000:

```
depsim start -port 9000
```

To start the server, have it listen on port 9000, log activity to /tmp/depsim.log and override the default configuration with settings from two configuration files:

```
depsim start -port 9000 -output /tmp/depsim.log
depsim_config1.json depsim_config2.json
```

To command a server running on the local host and listening on port 9000 to create an additional 100 device records of type IPAD and IPHONE:

```
depsim createdevice -port 9000 -count 100 IPAD IPHONE
```

To command a server running at myhost.example.com and listening on port 9000 to output its current device records and save them to the file at path /tmp/depsim_devices.json:

```
depsim dumpdevices -host myhost.example.com -port 9000 -output /
tmp/depsim_devices.json
```

To command a server running on the local host and listening on port 9000 to load an additional configuration file named depsim_config3.json:

```
depsim configure -port 9000 depsim_config3.json
```

To command a server running on the local host and listening on port 9000 to delete devices with serial numbers A34GHY432 and B54JY6342:

```
depsim deletedevices -port 9000 A34GHY432 B54JY6342
```

To command a server running on the local host and listening on port 9000 to delete all current device records:

```
depsim deletedevices -port 9000 -all
```

To command a server running on the local host and listening on port 9000 to revert to the originally specified configuration but not clear currently active session tokens:

```
depsim reset -port 9000 -preserve_sessions
```

Configuration Format

Configuration settings are specified in JSON format.

Configuration Categories

Account and credential settings

Account fields such as facilitator ID, organization address, and server name can be specified as well as OAuth fields such as access secret and access token.

Device records

Dynamically generated or pre-defined device records can be added to a server on a recurring or one time basis. Randomly chosen or pre-defined device records can also be deleted from a server on a recurring or one time basis.

API responses

Responses for each API can be configured. Any sequence of valid responses for each API can be specified, including of course the expected response to an API.

Configuration Keys

"server_token": <dictionary>

An optional dictionary of key-values that define the OAuth credentials accepted by the server.

Valid keys:

"access_secret": <string>

The default value is "AS_c31afd7a09691d83548489336e8ff1cb11b82b6bca13f793344496a556b1f4972eaff4dde6deb5ac9cf076fdfa97ec97699c34d515947b9cf9ed31c99dded6ba".

"access_token": <string>

The default value is "AT_927696831c59ba510cfe4ec1a69e5267c19881257d4bca2906a99d0785b785a6f6fdeb09774954fdd5e2d0ad952e3af52c6d8d2f21c924ba0caf4a031c158b89".

"consumer_key": <string>

The default value is "CK_48dd68d198350f51258e885ce9a5c37ab7f98543c4a697323d75682a6c10a32501cb247e3db08105db868f73f2c972bdb6ae77112aea803b9219eb52689d42e6".

"consumer_secret": <string>

The default value is "CS_34c7b2b531a600d99a0e4edcf4a78ded79b86ef318118c2f5bcfee1b011108c32d5302df801adbe29d446eb78f02b13144e323eb9aad51c79f01e50cb45c3a68".

"account": <dictionary>

An optional dictionary of one or more key-values that define the expected values to be returned in response to a "GET /account" request.

Valid keys:

"server_name": <string>

An optional string that is an identifiable name for the MDM server. The default value is "Example Server".

"server_uuid": <string>

An optional string that is the system-generated server identifier. The default value is "677cab70-fe18-11e2-b778-0800200c9a66".

"facilitator_id": <string>

An optional string that, in the production service, is the Apple ID of the person who generated the current tokens that are in use. The default value is "facilitator@example.com".

"org_name": <string>

An optional string that is the organization name. The default value is "Example Inc".

"org_email": <string>

An optional string that is the organization email address. The default value is "orgadmin@example.com".

"org_phone": <string>

An optional string that is the organization phone. The default value is "111-222-3333".

"org_address": <string>

An optional string that is the organization address. The default value is "123 Main St. Anytown, USA".

"admin_id": <string>

An optional string that is the Apple ID of the admin who generated the current token. The default value is "admin@example.com".

"org_type": <string>

An optional string that is the organization type ("edu" or "org"). The default value is "edu".

"org_version": <string>

An optional string that is the organization version. The default

value is "v1".

"device_insertions": <dictionary>

An optional dictionary specifying the rules for the creation and insertion of new device records or the insertion of pre-defined device records into the server. The dictionary can be used to cause one of two device insertion behaviors. The first behavior is to insert a list of pre-defined device records. The second behavior is to insert a dynamically generated set of device records. Additional keys allow for insertion of the records to occur on a repeating interval up to an optional iteration count.

Valid Keys:

"device_list": <array of dictionaries>

An optional array of device record dictionaries. Specify this key to add a list of pre-defined device records to the server. If this key is not specified then device records are dynamically generated. See section DEVICE RECORD KEYS for a list of key-values for a device record dictionary.

"device_types": <array of strings>

An optional array of fixed string values representing the types of the devices for the device records that are dynamically generated. The valid values are "IPAD", "IPHONE", "IPOD", "MAC". "device_types" is only applicable when device records are generated dynamically. Consequently, if "device_list" is specified, "device_types" is ignored.

"devices_per_event": <number>

An optional positive integer specifying the number of devices to add per iteration. Use this key in combination with "interval_in_seconds" to repeatedly add devices. If "device_list" is specified then the device records to add are taken from this list otherwise the device records are generated dynamically.

"interval_in_seconds": <number>

An optional non-negative integer specifying the number of seconds to wait between each operation of adding device records to the server. The number of device records to add per iteration is specified in "devices_per_event".

"max_event_count": <number>

An optional positive integer specifying the number of device insertion operations to perform. If "device_list" is specified, the total number of devices inserted is the lesser of either the count of devices in "device_list" or the value of "devices_per_event" multiplied by "max_event_count". If neither "device_list" nor "max_event_count" is specified, dynamically generated device records will continually be added to the server according the values specified in "devices_per_event" and "interval_in_seconds". In this scenario, specify "max_event_count" to limit the total number of dynamically generated device records to add.

"initial_delay_in_seconds": <number>

An optional non-negative integer specifying the number of seconds to wait before adding the first set of device records to the server.

Note: In order to cause the server to add dynamically generated device records, "devices_per_event" must be specified. If no additional keys are specified then the total number of dynamically generated device records added to the server is equal to the value specified for "devices_per_event".

"device_deletions": <dictionary>

An optional dictionary specifying the rules for deleting device records from the server. The dictionary can be used to cause one of two device deletion behaviors. The first behavior is to delete a list of pre-defined device records. The second behavior is to randomly delete device records. Additional keys allow for deletion of records to occur on a repeating interval up to an optional iteration count.

Valid Keys:

"device_list": <array of dictionaries>

An optional array of device record dictionaries. Specify this key to delete a list of pre-defined device records from the server. If this key is not specified then randomly selected device records are deleted. Only the "serial_number" key-value must be specified in a device record dictionary. All other key-values are ignored.

"devices_per_event": <number>

An optional positive integer specifying the number of devices to delete per iteration. Use this key in combination with "interval_in_seconds" to repeatedly delete devices. If "device_list" is specified then the device records to delete are taken from this list otherwise the device records are selected randomly.

"interval_in_seconds": <number>

An optional non-negative integer specifying the number of seconds to wait between each operation of deleting device records from the server. The number of device records to delete per iteration is specified in "devices_per_event".

"max_event_count": <number>

An optional positive integer specifying the number of device deletion operations to perform. If "device_list" is specified, the total number of devices deleted is the lesser of either the count of devices in "device_list" or the value of "devices_per_event" multiplied by "max_event_count". If neither "device_list" nor "max_event_count" is specified, randomly selected device records will continually be deleted from the server according the values specified in "devices_per_event" and "interval_in_seconds". In this scenario, specify "max_event_count" to limit the total number of randomly selected device records to delete.

"initial_delay_in_seconds": <number>

An optional non-negative integer specifying the number of seconds to wait before deleting the first set of device records from the server.

Note: In order to cause the server to delete randomly selected device records, "devices_per_event" must be specified. If no additional keys are specified then the total number of randomly selected device records deleted from the server is equal to the value specified for "devices_per_event".

"get_session_responses",
"get_account_responses",
"post_server_devices_responses",
"post_devices_sync_responses",
"post_devices_responses",
"post_devices_disown_responses",
"post_profile_responses",
"put_profile_devices_responses",
"get_profile_responses",
"delete_profile_devices_responses": <array of dictionaries>

An optional ordered array of one or more response dictionaries. A response dictionary contains a response value and the number of consecutive times to return that value. Each call to the server for the corresponding API will advance the server to the next specified response. The API corresponding to the key name can be derived from the key name itself. The first part of the key name corresponds to the API's HTTP method such as GET, POST, PUT, and DELETE. The second part of the key name corresponds to the API URL path following the host name. For example, "put_profile_devices_responses" corresponds to the API with method PUT and URL https://<hostname>/profile/devices.

Valid keys for all response dictionaries:

"count": <integer>

The number of consecutive times that the response should be sent

"response": <string>

The error response value. See section "VALID RESPONSES PER API" for a list of the valid values for each API. The value "expected_response" can be specified to cause the server to respond with the expected value for the request.

Additional key for put_profile_devices_responses, post_profile_responses, delete_profile_devices_responses, post_devices_responses, and post_devices_disown_responses only.

"device_list": <array of dictionaries>

An optional ordered array of one or more dictionaries. Each dictionary specifies a single device serial number. The specified response value in the response dictionary will only be returned for the matching device serial numbers in the original API request. The expected value is returned for all non-matching device

serial numbers. Key-values other than the device serial number are ignored. A "device_list" key-value can only specified if the value of "response" is "success", "not_accessible", "not_found", or "failed".

Additional key for get_session_responses only.

"retry_after": <string>

An optional string for the too_many_requests response for get_session_responses. The string value for this field can be either an HTTP-date or a positive integer number of seconds (in decimal) specifying the retry_after time returned in the response header. If no value is provided then a randomly generated retry_after value will be returned in the too_many_requests response.

Device Record Keys

"asset_tag": <string>

The asset tag value of the device.

"color": <string>

The color of the device.

"description": <string>

A description of the device.

"model": <string>

The model name of the device.

"serial_number": <string>

The serial number of the device.

Valid Responses Per API

"get_session_responses"

bad_request
forbidden_access_denied
forbidden_t_c_not_signed
invalid_method
unauthorized
too_many_requests

"get_account_responses"

forbidden
unauthorized
invalid_method

"post_server_devices_responses"
    bad_request_malformed_request_body
    bad_request_invalid_cursor
    bad_request_exhausted_cursor
    forbidden
    unauthorized
    invalid_method

"post_devices_sync_responses"
    bad_request_malformed_request_body
    bad_request_invalid_cursor
    bad_request_exhausted_cursor
    bad_request_cursor_required
    forbidden
    unauthorized
    invalid_method

"post_devices_responses"
    success
    not_found
    forbidden
    bad_request_malformed_request_body
    bad_request_device_id_required
    unauthorized
    invalid_method

"post_devices_disown_responses"
    success
    not_accessible
    not_found
    forbidden
    bad_request_malformed_request_body
    bad_request_device_id_required
    unauthorized
    invalid_method

"post_profile_responses"
    success
    not_accessible
    failed
    forbidden
    bad_request_malformed_request_body
    bad_request_config_url_required

bad_request_config_name_required
bad_request_department_required
bad_request_support_phone_required
unauthorized
invalid_method

"put_profile_devices_responses"
success
not_accessible
failed
forbidden
bad_request_malformed_request_body
bad_request_device_id_required
bad_request_profile_uuid_required
bad_request_profile_not_found
unauthorized
invalid_method

"get_profile_responses"
forbidden
bad_request_profile_uuid_required
bad_request_profile_not_found
unauthorized
invalid_method

"delete_profile_devices_responses"
success
not_accessible
failed
forbidden
bad_request_device_id_required
unauthorized
invalid_method

Configuration Examples

Example One:

This configuration specifies the following:

- Alternate account values for server_name, org_email, and org_name
- A device insertion rule to dynamically create and add ten device records of type IPAD and MAC every five seconds and performs this operation fifteen times.

- A custom response configuration to the PUT /profile/devices API that returns "FAILED" for all device records for the first two calls to the API, then returns the expected response once, then returns "NOT_ACCESSIBLE" for all device records for the next three calls, then returns the expected response on subsequent calls.

```
{
    "account": {
        "org_email": "john@acme.com",
        "org_name": "Acme Inc",
        "server_name": "Acme Server",
    },
    "device_insertions": {
           "device_types": [
                "IPAD",
                "MAC"
           ],
           "devices_per_event": 10,
           "interval_in_seconds": 5,
           "max_event_count": 15
     },
    "put_profile_devices_responses": [
        {
            "count": 2,
            "response": "failed"
        },
        {
            "count": 1,
            "response": "expected_response"
        },
        {
            "count": 3,
            "response": "not_accessible"
        }
     ]
}
```

Example Two:

This configuration specified the following:

- A custom response configuration to the GET /session API that returns an HTTP 403 error and "T_C_NOT_SIGNED" in the response body once, then returns the expected response on subsequent calls.
- A device insertion rule to add two pre-defined device records.
- A custom response configuration to the POST /devices API that returns an HTTP 400 error and "DEVICE_ID_REQUIRED" in the response body once, that returns "NOT_FOUND" for a device with serial number L1WIBUYQQA7EZHIA once, and returns the expected response on subsequent calls.

```
{
    "get_session_responses": [
        {
            "count": 1,
            "response": "forbidden_t_c_not_signed"
        },
        {
            "count": 1,
            "response": "too_many_requests"
        }
     ],
    "device_insertions": {
        "device_list": [
            {
                "asset_tag": "46189",
                "color": "SPACE GRAY",
                "description": "IPAD LTE 16GB",
                "model": "IPAD",
                "serial_number": "L1WIBUYQQA7EZHIA"
            },
            {
                "asset_tag": "31778",
                "color": "WHITE",
                "description": "IPAD WIFI 64GB",
                "model": "IPAD",
                "serial_number": "YOEWB28Z85SIWPQ9"
            }
        ],
     }
    "post_devices_responses": [
           {
            "count": 1,
            "response": "bad_request_device_id_required"
           },
           {
            "count": 1,
```

```
            "device_list": [
                        {
                              "serial_number": "L1WIBUYQQA7EZHIA"
                        }
            ],
            "response": "not_found"
           }
      ]
}
```

Example Three:

This configuration specifies settings for all supported keys and illustrates the complete structure of the configuration format. Note that not all settings specified in this example are intended to be used together. Refer to the documentation for each setting for details.

```
{
    "server_token": {
        "access_secret":
           "AS_c31afd7a09691d83548489336e8ff1cb11b82b6bca13f793344496a
           556b1f4972eaff4dde6deb5ac9cf076fdfa97ec97699c34d515947b9cf9
           ed31c99dded6ba",
        "access_token":
           "AT_927696831c59ba510cfe4ec1a69e5267c19881257d4bca2906a99d0
           785b785a6f6fdeb09774954fdd5e2d0ad952e3af52c6d8d2f21c924ba0c
           af4a031c158b89",
        "access_token_expiry": "2015-07-29T11:35:52-07:00",
        "consumer_key":
           "CK_48dd68d198350f51258e885ce9a5c37ab7f98543c4a697323d75682
           a6c10a32501cb247e3db08105db868f73f2c972bdb6ae77112aea803b92
           19eb52689d42e6",
        "consumer_secret":
           "CS_34c7b2b531a600d99a0e4edcf4a78ded79b86ef318118c2f5bcfee1
           b011108c32d5302df801adbe29d446eb78f02b13144e323eb9aad51c79f
           01e50cb45c3a68"
    },
    "account": {
        "facilitator_id": "facilitator@example.com",
        "org_address": "123 Main St. Anytown, USA",
        "org_email": "orgadmin@example.com",
        "org_name": "Example Inc",
        "org_phone": "111-222-3333",
        "server_name": "Example Server",
```

```
    "server_uuid": "677cab70-fe18-11e2-b778-0800200c9a66",
    "admin_id": "admin@example.com",
    "org_type": "edu",
    "org_version": "v1"
},
"device_insertions": {
    "device_list": [
        {
            "asset_tag": "46189",
            "color": "SPACE GRAY",
            "description": "IPAD LTE 16GB",
            "model": "IPAD",
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "asset_tag": "31778",
            "color": "WHITE",
            "description": "IPAD WIFI 64GB",
            "model": "IPAD",
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "device_types": [
        "IPAD",
        "IPHONE"
    ],
    "devices_per_event": 2,
    "initial_delay_in_seconds": 0,
    "interval_in_seconds": 2,
    "max_event_count": 1000
},
"device_deletions": {
    "device_list": [
        {
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "devices_per_event": 2,
    "initial_delay_in_seconds": 0,
    "interval_in_seconds": 2,
    "max_event_count": 1000
```

```
},
"get_session_responses": [
    {
        "count": 1,
        "response": "bad_request"
    },
    {
        "count": 1,
        "response": "forbidden_access_denied"
    },
    {
        "count": 1,
        "response": "forbidden_t_c_not_signed"
    },
    {
        "count": 1,
        "response": "invalid_method"
    },
    {
        "count": 1,
        "response": "unauthorized"
    },
    {
        "count": 1,
        "response": "too_many_requests"
    },
    {
        "count": 1,
        "response": "expected_response"
    }
],
"get_account_responses": [
    {
        "count": 1,
        "response": "forbidden"
    },
    {
        "count": 1,
        "response": "unauthorized"
    },
    {
        "count": 1,
        "response": "invalid_method"
    },
```

```
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "post_server_devices_responses": [
        {
            "count": 1,
            "response": "bad_request_malformed_request_body"
        },
        {
            "count": 1,
            "response": "bad_request_invalid_cursor"
        },
        {
            "count": 1,
            "response": "bad_request_exhausted_cursor"
        },
        {
            "count": 1,
            "response": "forbidden"
        },
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "post_devices_sync_responses": [
        {
            "count": 1,
            "response": "bad_request_malformed_request_body"
        },
        {
            "count": 1,
            "response": "bad_request_invalid_cursor"
        },
```

```
        {
            "count": 1,
            "response": "bad_request_exhausted_cursor"
        },
        {
            "count": 1,
            "response": "bad_request_cursor_required"
        },
        {
            "count": 1,
            "response": "forbidden"
        },
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "post_devices_responses": [
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
                }
            ],
            "response": "success"
        },
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
```

```
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
                }
            ],
            "response": "not_found"
        },
        {
            "count": 1,
            "response": "forbidden"
        },
        {
            "count": 1,
            "response": "bad_request_malformed_request_body"
        },
        {
            "count": 1,
            "response": "bad_request_device_id_required"
        },
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "post_devices_disown_responses": [
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
                }
            ],
            "response": "success"
        },
```

```
{
    "count": 1,
    "device_list": [
        {
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "response": "not_accessible"
},
{
    "count": 1,
    "device_list": [
        {
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "response": "failed"
},
{
    "count": 1,
    "response": "forbidden"
},
{
    "count": 1,
    "response": "bad_request_malformed_request_body"
},
{
    "count": 1,
    "response": "bad_request_device_id_required"
},
{
    "count": 1,
    "response": "unauthorized"
},
{
    "count": 1,
    "response": "invalid_method"
},
```

```
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "post_profile_responses": [
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
                }
            ],
            "response": "success"
        },
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
                }
            ],
            "response": "not_accessible"
        },
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
                }
            ],
            "response": "failed"
        },
        {
            "count": 1,
```

```
            "response": "forbidden"
        },
        {
            "count": 1,
            "response": "bad_request_malformed_request_body"
        },
        {
            "count": 1,
            "response": "bad_request_config_url_required"
        },
        {
            "count": 1,
            "response": "bad_request_config_name_required"
        },
        {
            "count": 1,
            "response": "bad_request_department_required"
        },
        {
            "count": 1,
            "response": "bad_request_support_phone_required"
        },
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "put_profile_devices_responses": [
        {
            "count": 1,
            "device_list": [
                {
                    "serial_number": "L1WIBUYQQA7EZHIA"
                },
                {
                    "serial_number": "YOEWB28Z85SIWPQ9"
```

```
            }
        ],
        "response": "success"
    },
    {
        "count": 1,
        "device_list": [
            {
                "serial_number": "L1WIBUYQQA7EZHIA"
            },
            {
                "serial_number": "YOEWB28Z85SIWPQ9"
            }
        ],
        "response": "not_accessible"
    },
    {
        "count": 1,
        "device_list": [
            {
                "serial_number": "L1WIBUYQQA7EZHIA"
            },
            {
                "serial_number": "YOEWB28Z85SIWPQ9"
            }
        ],
        "response": "failed"
    },
    {
        "count": 1,
        "response": "forbidden"
    },
    {
        "count": 1,
        "response": "bad_request_malformed_request_body"
    },
    {
        "count": 1,
        "response": "bad_request_device_id_required"
    },
    {
        "count": 1,
        "response": "bad_request_profile_uuid_required"
    },
```

```
        {
            "count": 1,
            "response": "bad_request_profile_not_found"
        },
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "get_profile_responses": [
        {
            "count": 1,
            "response": "forbidden"
        },
        {
            "count": 1,
            "response": "bad_request_profile_uuid_required"
        },
        {
            "count": 1,
            "response": "bad_request_profile_not_found"
        },
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ],
    "delete_profile_devices_responses": [
```

```
{
    "count": 1,
    "device_list": [
        {
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "response": "success"
},
{
    "count": 1,
    "device_list": [
        {
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "response": "not_accessible"
},
{
    "count": 1,
    "device_list": [
        {
            "serial_number": "L1WIBUYQQA7EZHIA"
        },
        {
            "serial_number": "YOEWB28Z85SIWPQ9"
        }
    ],
    "response": "failed"
},
{
    "count": 1,
    "response": "forbidden"
},
{
    "count": 1,
    "response": "bad_request_device_id_required"
},
```

```
        {
            "count": 1,
            "response": "unauthorized"
        },
        {
            "count": 1,
            "response": "invalid_method"
        },
        {
            "count": 1,
            "response": "expected_response"
        }
    ]
}
```