iRMC RESTful Tools v6

This document describes a toolset for changing profiles for iRMC and BIOS, installing operating systems, uploading and starting custom images and other features via the RESTful API for PRIMERGY servers.

Any feedback is appreciated.

IMPORTANT	**Please note: This tool set is provided W/O ANY WARRANTY and usage at your own risk!** == History .ChangeLog

Version	Date	Author	Changes
6	2023-03-28	Jürgen Orth	+ Security enhancement for `resttool` + Some corrections in README
5	2023-03-17	Justin Langosch	- Changed format to ASCIIdoc and updated links
4	2020-08-13	Jürgen Orth	* Extension with scripts for OS deployment and a few more tools
3	2020-07-23	Jürgen Orth	* Extension by scripts for handling custom images: upload, list, delete, boot
2	2019-07-17	Jürgen Orth	+ Description new option "-n" to prevent reboots when setting profile settings. Notes regarding monitoring in the iRMC.
1	2018-11-01	Jürgen Orth	- Initial Version

Legend:
* New feature
+ Improvement
! Change that may require modification
x Bugfix
- Information

Table of Contents

Introduction

Base Management Controllers are components of x86 servers that allow out-of-band management of hardware. The implementation used by Fujitsu is called Integrated Remote Management Controller (hereinafter referred to as iRMC). All of these management controllers have a web interface for interactive operation. iRMC also allows automated operation via a RESTful API from the S4 generation onwards [API]. The scripts described here use this API for profile management, custom image management, operating system deployment and various help functions.

The programs and tools described here are not official Fujitsu software components. They are only an example implementation of the official RESTful API Specification [API] and the descriptions of the Profile Specification [ProfileSpec].

WARNING	The use is therefore at your own risk and without any guarantee! Please also note the hints in chapter "Reboot behaviour"!

Requirements

Hardware

On the hardware side, an iRMC of generation S4 or higher is required on which the RESTful protocol is available and enabled. The user used for authentication must have the permission [GUI] to use the RESTful API [API] order. From iRMC Generation S5 on, this is already the delivery status. For the S4 generation the following screenshot shows the necessary settings:
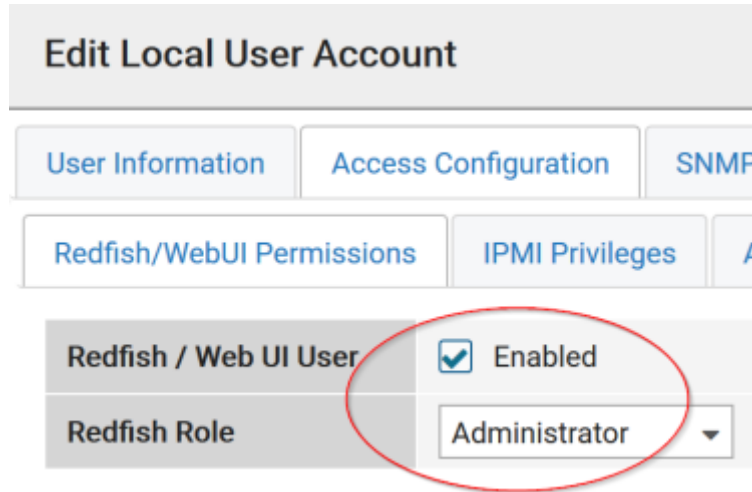
The examples shown here were performed on a PRIMERGY RX2540 M4 with iRMC version 1.25P and BIOS version R1.22.0 to iRMC version 2.6x and BIOS version R1.4x, and on a PRIMERGY RX2540 M5 with iRMC version 2.63P and BIOS version R1.19.0. In addition, the eLCM (embedded Lifecycle Management) license is required to use the Custom Image and OS Deployment functions

Software

On the software side, the free tool cURL is necessary, which handles the data transfer between program and iRMC via RESTful API [API] done. The scripts described here are BASH scripts and can therefore be used natively under all UNIX/Linux environments. For use under Windows, its Linux subsystem can be used. Alternatively, other tools like Cygwin are also suitable

Installation

If you've git you simply install this tool by 'git clone https://github.com/fujitsu/iRMC-REST-API'. Afterwards the tools is located in directory iRMC-REST-API. Please extend your PATH var to this directory.

```
$ git clone https://github.com/fujitsu/iRMC-REST-API
Cloning into 'iRMC-REST-API'...
remote: Enumerating objects: 45, done.
remote: Counting objects: 100% (45/45), done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 45 (delta 9), reused 35 (delta 5), pack-reused 0
Receiving objects: 100% (45/45), 889.12 KiB | 5.11 MiB/s, done.
Resolving deltas: 100% (9/9), done.
```

```
    $ export PATH=$PATH:$PWD/iRMC-REST-API

    $ ls -F iRMC-REST-API
    LICENSE           example/          restgetmac*      restresetirmc*
    README.adoc       restbootci*       restgetprofile*  restsetdeploy*
    chasset*          restcheckeim*     restgetversion*  restsetprofile*
    chpw*             restdeleteci*     restlistci*      resttool
    deploy/           restdeployos*     restlistos*      restuploadci*
    docs/             restgetbootmac*   restloadeim*
    downloadserver/   restgetdeploy*    restoptions
```

Otherwise download and unpack this zipfile in any directory. It is recommended to do this in a directory which is contained in the PATH variable (about $HOME/bin). Alternatively, you can extend the PATH variable with any directory of your own.

Script Overview

Now you will find the following files in the iRMC-REST-API directory:

- README.adoc
  This document.

- doc/README.pdf
  This document in pdf format.

- img/*
  Some included images of README file.

- restoptions
  Contains (among others) the default settings for iRMC address, user data etc.

```bash
#!/bin/bash
#
# restoptions: tools for using RESTful API with Fujitsu iRMC S4/S5
#
# (c) Juergen Orth - Fujitsu Technology Solutions GmbH
#
# $Id: restoptions 174 2023-03-28 14:08:48Z HMBJOrth $


########################################################################
#
# Default-Settings
# ================
#
# Can/should be adjusted according to user preferences
#

########################################################################
# iRMC settings
```

```
        IRMC=${IRMC:-10.172.124.82}

        # USER: iRMC User with appropriate rights (default: administrator)
        USER=${IRMC_CRED%:*}
        USER=${USER:-admin}                      # if iRMC_CRED not set use defaults

        # PD: Password for above USER (default: admin)
        PW=${IRMC_CRED#*:}
        PW=${PW:-${DCDEFAULTPW:-admin}} # if iRMC_CRED not set use defaults

        # Profile settings
        DEFAULTPROFILE=IrmcConfig/BootConfig
        JSON=profile.json               # Filename of profile json string
        POSTOPTION=""                   # Option for Post (inhibit reboot)

        # Custom image settings
        CUSTOMIMAGEURLPATH="http://10.172.125.100/DownloadManager/globalflash/iso"
        CUSTOMIMAGENAME=customimage
        CUSTOMIMAGESUFFIX="iso"

        # Deploy settings
        DEFAULTOSID=3
        DEFAULTOS=deploy/esxi67u3_IDE_m2.json

        # Reset options
        RESETWAITTIME=10                # Waittime before performing reset

        # Debug settings
        DEBUGLEVEL=0                    # Debuglevel: 0=nothing, 1=low, 2=medium,
 3=high
```

This data can be adapted to your own environment to save you typing work. This script is not called directly by the user. However, almost all parameters can be passed via command line parameters of the respective user commands, which then override these defaults.

- resttool
  This contains various library routines. This script is therefore only included by other scripts and is not directly executed by the user.

- rest* (e.g., restgetprofile/restuploadci/restdeployos/restgetmac/restresetirmc/⋯)
  These scripts that can be called by the user are described in the corresponding chapters. It may be advisable to set the PATH variable to include the directory where these tools were unpacked. In general, an online help of possible parameters can be displayed for all commands via the switch "-h". See also the example below.

| WARNING | Please consider the Security concerns regarding using user name and password as commandline parameters. |

- example
  Directory with a sample CSV file containing some server info.

- download server

  Directory with sample XML files for the configuration of the download server.

- deploy

  Directory with a few example JSON files for the installation/configuration (deploy) of various operating systems.

Display of the possible commandline parameters:

```
        $ restgetprofile -h
        Usage: restgetprofile [-i <iRMC name/IP> (10.172.126.245)] [-u <user> (admin)]
[-p
        <password> (admin)] [-f <filename> (profile.json)] [-d <debuglevel> (0)] [-h
(help)]
        [<profilename> (IrmcConfig/BootConfig)]
```
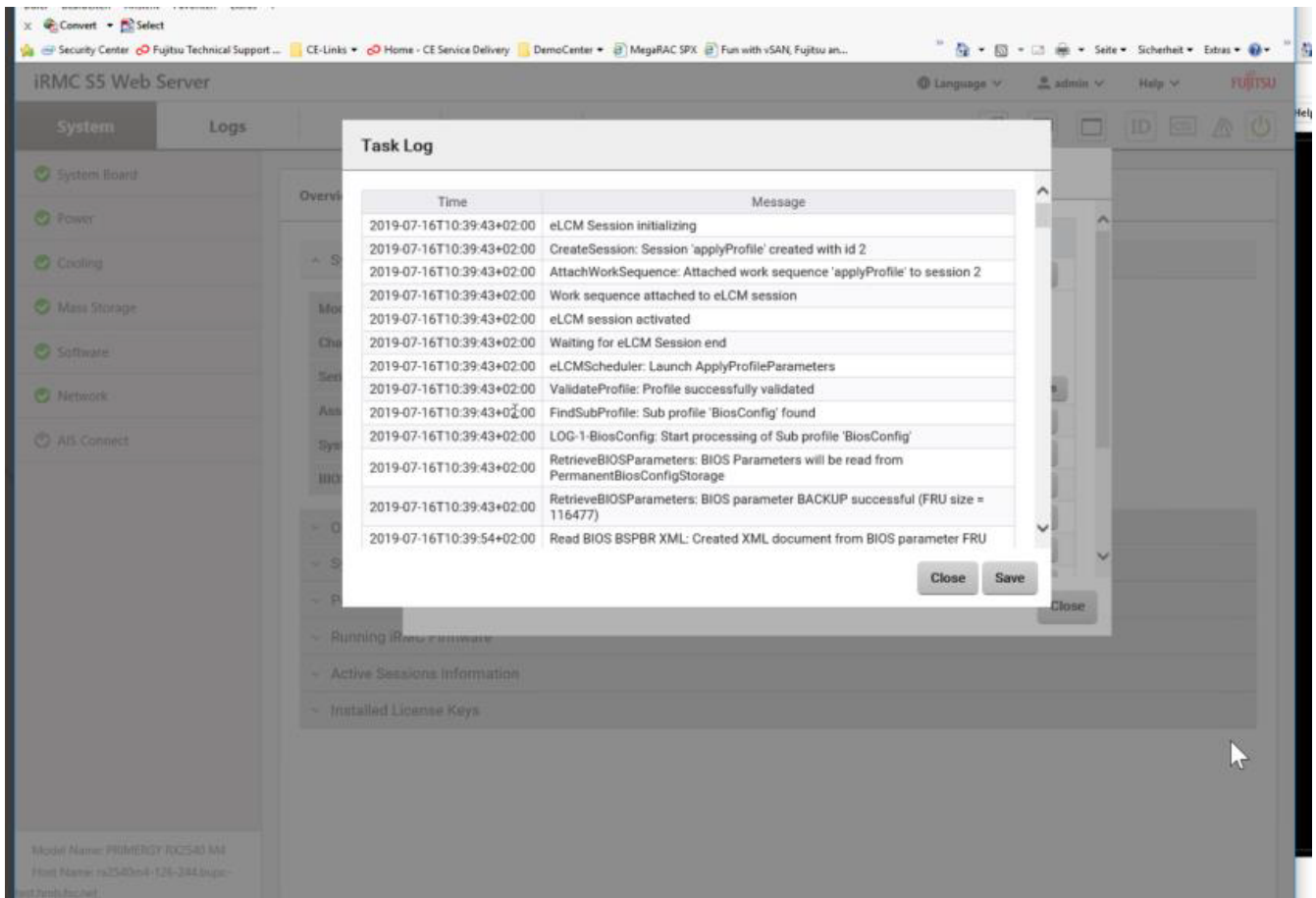
This help function can be applied to all scripts. If individual switches are not specified, the default settings of the file restoptions are used. The default settings are shown in round brackets.

Monitoring

Some functions that are called via the RESTful API are asynchronous in the background. This basically concerns all operations that may take longer, such as uploading an image with many megabytes of data. For this purpose iRMC-internal "sessions" are created. These sessions can be tracked in the iRMC (from FW version 2.20) in the "Task Manager":

Profile Management

General operation

The different profile settings [ProfileSpec] are stored in a tree structure. There is one for the BIOS and one for the iRMC. These profiles can be read and written completely (e.g. IrmcConfig) or partially (e.g. IrmcConfig/System/AssetTag).

Download

The settings (profiles) stored in the BIOS or the iRMC can or must first be copied to an iRMC-internal buffer (store) before the actual download. The store must not yet contain a profile with the same name! This copying process is performed asynchronously in one session. Therefore you have to wait until this session is terminated. Afterwards, they can be transferred from this store to the computer of the calling CLI program. All these steps are done by the `restgetprofile` script.

Upload

The upload takes place without a detour via a buffer (store) but also asynchronously, so that the end of the session must be waited for. The process can therefore take a few seconds. The upload is done using the script `restsetprofile`.

General procedure

For an automated profile change, it should first be ensured that all affected servers are of the same type and have the same BIOS/iRMC-FW versions, since different versions may also have different parameters added or removed. Once a "master profile" has been read out via `restgetprofile`, it can be adapted according to your requirements and distributed to all servers of the same type with `restsetprofile`.

Reboot behaviour

IMPORTANT	**Depending on the iRMC settings, a reboot is triggered already when reading BIOS profile settings! To prevent this, the iRMC setting "Enable Automatic BIOS Parameter Backup" must be activated:**

If this setting is activated, then a writing of BIOS-settings without immediate reboot can optionally be performed. This is enabled by the "-n" (noreboot) option in the `restsetprofile` command. The BIOS changes are of course only active after the next reboot of the server.

This option is not required for changes to iRMC settings. There is no server reboot during reading or writing and changes are effective immediately.

Profile Management Scripts

- `restgetprofile profiles`
  This script reads a profile from the iRMC (iRMC- or BIOS-Settings) starting with the level IrmcConfig or BiosConfig according to chapter 5 [API]. A call with the switch -h returns a short Usage.

- `restset profiles`
  This script transfers a profile from a file to the iRMC (iRMC or BIOS settings). A call with switch -h returns a short Usage.

- `chasset`
  A script to directly modify the asset tag. It uses the command `restsetprofile`.
  Example call: `chasset InvNo 47110815`

  Since this simple script changes the asset tag string via sed, special characters such as "/" may have to be quoted accordingly, i.e. "/" must be changed to "\\/".

Example session

This example shows how to configure a separate NTP server for the iRMC. It is recommended to make a backup of all settings first

IMPORTANT	**Please read the notes about reboot behavior!**

```
$ restgetprofile -f BIOSbackup.json BiosConfig +
$ restgetprofile -f iRMCbackup.json IrmcConfig
```

Now all BIOS settings are stored in BIOSbackup.json. The same applies to the iRMC settings.

Read out the time settings of the iRMC:

```
$ restgetprofile IrmcConfig/Time
2018-09-03 11:08:36 -- retrieving IrmcConfig/Time +
2018-09-03 11:08:37 -- Reading profile - be patient ... +
.. +
2018-09-03 11:08:48 -- Profiles stored in profile.json +
2018-09-03 11:08:48 -- Cleaning up
```

and display of the generated file (here profile.json):

```
$ cat profile.json
{
"Server":{
    "SystemConfig":{
        "IrmcConfig":{
            "@Processing": "execute",
```

```
                "Time":{
                    "Ntp":{
        "Servers":{
                        "Server":[
                            {
                            "@ServerIdx":0,
                            "HostName": "time.fsc.net"
                            },
                            {
                            "@ServerIdx":1,
                            "HostName": "pool.ntp.org"
                            }
                            ]
                            }
                    },
                    "TimeZoneOffset":0,
                    "SyncSource": "NTP",
                    "TimeZone": "Europe\/Berlin
                    "RtcMode": "LocalTime",
                    "MmbTimeSyncEnabled":true
                    },
                    "@Version": "1.04"
        }

                    },
                "@Version": "1.01"
            }
        }
```

Change the entry for the second NTP server (i.e. from `pool.ntp.org` to `ntp.mycompany.com`):

`$ sed -i 's/pool.ntp.org/ntp.mycompany.de/' profile.json`

Upload the modified profile:

```
$ restsetprofile profile.json +
2018-09-03 11:10:02 -- Assigning profile profile.json +
2018-09-03 11:10:03 -- Waiting for session to terminate - be patient ... +
.......... +
2018-09-03 11:10:45 -- Cleaning up
```

For example, if you have a CSV file `myassetlist.csv` (with the separator ";"), which lists the server names in column 1 and the names or IP addresses of the corresponding iRMC in column 2, you can use the following lines to provide all servers with new settings in one step. Logs are also stored for each server in the /tmp directory.

```
$ # Display myasssetlist.csv
$ cat example/myassetlist.csv
server1;irmc1;RZ2;Rack 1;admin;secret1;admin meier;
server2;192.168.47.11;RZ1;Rack 4;admin;secret2;admin mueller;
```

```
        server3;irmc3.mydomain.de;RZ2;Rack 2;admin;secret3;admin meier;

        $ IFS=';'
        $ while read server irmc raum rack user password somethingmore
        > do
        > echo "Setting profile profile profile.json for server $server / iRMC=$irmc
        > restsetprofiles -i $irmc -u $user -p "$password" profile.json 2>&1 | tee
        /tmp/$server.log
        > done < example/myassetlist.csv
```

Custom Image Handling

The iRMC, in conjunction with licensed eLCM, is able to boot bootable images that have been uploaded to the internal SD card. This makes it possible, for example, to start ISO images created by Content Collector/Update DVD with updates fully automatically and to update the system accordingly.

IMPORTANT	The following instructions are mandotry for Custom Image Handling!

- Uploading an image requires an http or https URL with the boot image as source.

- The image file **must** have the extension `.iso.`

- Booting a custom image performs an **immediate reset of the server**. So you should make sure that the server was either turned off or booted into the BIOS before. The latter can also be done automatically via IPMI.

  - Switch on the BIOS via IPMI One-Boot (`ipmitool [options···] chassis bootparam set bootflag force_bios`).

  - OS reboot (ssh/WMI)

- `restuploadci [<Image_URL>]`
  This script loads an image file to the internal SD card of the eLCM. If no switches for iRMC name/IP address, User, Password, Image_URL are given, the script uses default values of the file `restoptions`. If URL path or extension are omitted, they are automatically supplemented with the default settings from `restoptions`. The upload takes about 5 to 30 minutes depending on the image size. Please note that both the total size (~8GB) for images on the SD card and the number (7) of possible images are limited.

- `restlistci`
  This script lists the already existing images on the internal SD card. If no switches for iRMC name/IP address, user, password are given, the script uses default values of the file `restoptions`.

- `restdeleteci [<ImageName>]`
  This script deletes the specified image on the internal SD card. If no switches for iRMC name/IP address, user, password are given, the script uses default values of the file `restoptions`.

- `restbootci [<ImageName>]`
  This script boots the specified image. A possibly running OS will be reset immediately! If no switches for iRMC name/IP address, user, password are given, the script uses default values of the file `restoptions`.

Example session

Default values of the file `restoptions`. Can/should be adjusted accordingly.

```
$ grep -i custom restoptions
CUSTOMIMAGEURLPATH="http://10.172.125.100/DownloadManager/globalflash/iso"
CUSTOMIMAGENAME=customimage
CUSTOMIMAGESUFFIX="iso"
```

Listing of already installed images:

```
$ restlistci
UPDATE_DVD_irmc_down_manual
VMware_ESXi_7.0.0_15843807_Fujitsu_v500_1
```

Upload an image:

```
$ restuploadci
 http://fjrepo1/DownloadManager/globalflash/iso/UPDATE_DVD_irmc_up_auto.iso
        2020-07-23 11:12:38 -- Uploading
 http://fjrepo1/DownloadManager/globalflash/iso/UPDATE_DVD_irmc_up_auto.iso
        to 10,172,124,225
        .....................
        2020-07-23 11:15:01 -- Done
        2020-07-23 11:15:01 -- Cleaning up
```

Booting an image <mark>(PLEASE PAY ATTENTION TO ABOVE WARNING !):</mark>

```
$ restbootci UPDATE_DVD_irmc_up_auto
        2020-07-23 11:15:43 -- Booting >UPDATE_DVD_irmc_up_auto< - this can take a
couple of minutes ...
        ........................................
        2020-07-23 11:20:16 -- Done
        2020-07-23 11:20:17 -- Cleaning up
```

Delete an image:

```
$ restdeleteci UPDATE_DVD_irmc_up_auto
        2020-07-23 12:37:24 -- Deleting custom image >UPDATE_DVD_irmc_up_auto< ...
        2020-07-23 12:37:27 -- Done
```

OS Deploy

In conjunction with licensed eLCM, the iRMC is able to install (deploy) operating systems supported by the embedded Installation Manager (eIM) and to configure them extensively (network, user, disk, local settings, …). The operating system image to be installed is made available to the iRMC via an http(s) URL. Configuration parameters are stored in a separate, local JSON file for [ProfileSpec] deposit. Furthermore, access to a web download server for installation tool SVIM is required. This server is either provided on the Internet by Fujitsu or by using the Fujitsu Repository Server on the local customer network, which is practically a mirror of the Fujitsu server.

Management of the web download server:

- restgetdeploy
  This script lists the current configuration of the iRMC regarding the web download server in JSON format.

```
$ restgetdeploy
2020-08-13 07:47:50 -- Reading Web download server settings on 10.172.124.225
...
{
    "Repository":{
        "Server":{
            "URL": "https://webdownloads.ts.fujitsu.com",
            "UseProxy": "yes"
        },
        "Proxy":{
            "URL":"172.25.161.100",
            "Port": "81",
    "User":"(zero)"
            "Password": "*****"
        }
    }
}
```

- restsetdeploy <deploy-config.xml>
  This script adjusts the settings (URL/Proxy) of the web download server according to the entries in the **XML file**. The tarball contains sample configs in the subdirectory ./downloadserver. If you use your own repository server, make sure that the option RepositoryMode has the value 1 (=eIM repository only) or 2 (=globalflash and eIM repository). This parameter is set in the file /opt/fujitsu/ServerViewSuite/RepServer/cfg/RepServer.conf

The download server settings can also be made in the iRMC GUI in the Settings/Service area:
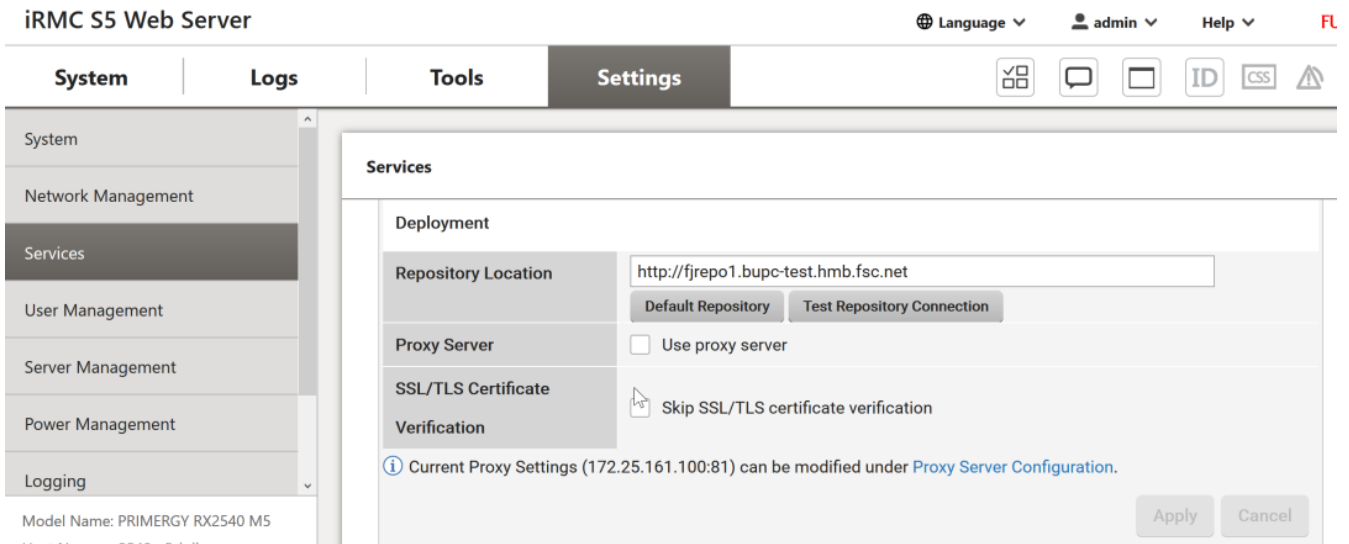
Loading the OS catalog:

- `restlistos`
  This script updates the list of supported operating systems by updating and displaying the catalog from the web download server. The displayed index value for a specific OS is needed in the next step.

Update of the eIM:

- `restloadeim <OSTypeId>`
  This script copies the current embedded Installation Manager (eIM) to the SD card of the eLCM, if a current version is not already available. Depending on the network speed, an upload can take much longer than one hour.

Installing the operating system:

- `restdeployos <OS-config.json>`
  This script installs the operating system defined in the specified JSON file according to the + config parameters also stored. The URL for the OS image is also stored in this JSON file.

WARNING	**Calling this script will reset the server immediately!**

The syntax for the JSON file is described in the manual "Profile Specification [ProfileSpec] documented. The URL for the OS mentioned in the JSON file must contain an unpacked ISO:

```
...
"OsRemotePath": "ftp://10.172.124.108/iso-custom/esxi67u3",
...
```
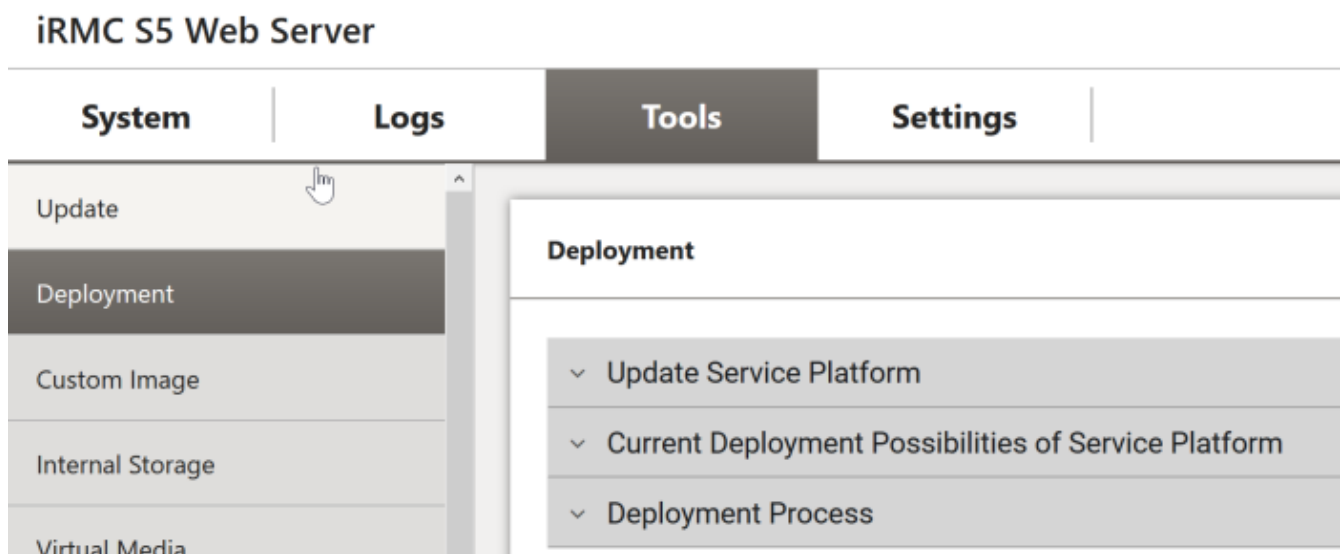
Index of ftp://10.172.124.108/iso-custom/esxi67u3/

📁 Up to higher level directory

Name	Size	Last Modified	
File: .DISCINFO	1 KB	9/24/2019	5:28:00 AM
File: ATA_LIBA.V00	96 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V00	9 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V01	7 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V02	7 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V03	8 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V04	9 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V05	8 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V06	7 KB	9/24/2019	5:28:00 AM
File: ATA_PATA.V07	8 KB	9/24/2019	5:28:00 AM
File: B.B00	128 KB	9/24/2019	5:28:00 AM
File: BLOCK_CC.V00	25 KB	9/24/2019	5:28:00 AM
File: BNXTNET.V00	146 KB	9/24/2019	5:28:00 AM

The installation time depends on the operating system, network speed and config options. It can be assumed from about 30 to 120 minutes.

The corresponding GUI view is:

iRMC S5 Web Server

System	Logs	Tools	Settings	

Update

Deployment

Custom Image

Internal Storage

Virtual Media

Deployment

- ⌄ Update Service Platform
- ⌄ Current Deployment Possibilities of Service Platform
- ⌄ Deployment Process

Example session

```
$ time restlistos
2020-08-06 11:46:11 -- Updating catalog ...
2020-08-06 11:46:15 -- Displaying catalog ...
Platform version: 13.19.12.10
1 Microsoft Hyper-V Server 2019
2 Microsoft Hyper-V Server 2016
3 VMware ESXi 6.7 update 3
4 VMware ESXi 6.7 update 2
5 VMware ESXi 6.7 update 1
6 VMware ESXi 6.5 update 3
```

```
   7 VMware ESXi 6.5 update 2
   8 SUSE Linux Enterprise Server 15 (x86_64) update 1
   9 SUSE Linux Enterprise Server 12 (x86_64) update 5
  10 SUSE Linux Enterprise Server 12 (x86_64) update 4
  11 Red Hat Enterprise Linux 8 (x86_64) update 1
  12 Red Hat Enterprise Linux 8 (x86_64)
  13 Red Hat Enterprise Linux 7 (x86_64) Compute Node update 7
  14 Red Hat Enterprise Linux 7 (x86_64) Compute Node update 6
  15 Red Hat Enterprise Linux 7 (x86_64) update 7
  16 Red Hat Enterprise Linux 7 (x86_64) update 6
  17 Windows Server Datacenter, version 1909
  18 Windows Server Standard, version 1909
  19 Windows Server Datacenter, version 1903
  20 Windows Server Standard, version 1903
  21 Windows Server 1809 Datacenter
  22 Windows Server 1809 Standard
  23 Windows Server 2019 Essentials
  24 Windows Server 2019 Datacenter
  25 Windows Server 2019 Standard
  26 Windows Server 1803 Datacenter
  27 Windows Server 1803 Standard
  28 Windows Server 1709 Datacenter
  29 Windows Server 1709 Standard
  30 Windows Storage Server 2016 Standard
  31 Windows Server 2016 Essentials
  32 Windows Server 2016 Datacenter
  33 Windows Server 2016 Standard

  real0m7 .491s
  user0m0 .630s
  sys0m1 .769s

  $ time restloadeim 3
  2020-08-06 11:46:36 -- Uploading eIM image with OsTypeId 3. This can take a
very long time ...
  2020-08-06 11:46:40 -- Waiting for session 1 to terminate - be patient ...
  ..
  2020-08-06 11:46:52 -- Session 1 finished with status: terminated regularly
  2020-08-06 11:46:52 -- Cleaning up

  real0m18 .558s
  user0m1 ,411s
  sys0m4 .980s

  $ time restdeployos deploy/esxi67u3.json
  SVIM version 13.19.12.10 found.
  2020-08-06 11:47:13 -- Assigning profile deploy/esxi67u3.json
  2020-08-06 11:47:18 -- Waiting for session 1 to terminate - be patient ...
  .......................................
  2020-08-06 12:10:50 -- Session 1 finished with status: terminated regularly
  2020-08-06 12:10:50 -- Cleaning up
```

```
        real23m43 ,570s
        user1m27 ,406s
        sys5m18 .964s
```

Other Tools

The toolset also contains the following tools: * `restresetirmc`
Reboot the iRMC. The process can be aborted within 10 seconds (or in `restoptions` according to RESETWAITTIME) by SIGINT (Ctrl-C). The server must not be in the BIOS.

- `restgetmac`
  Lists the MAC addresses of all NICs. The server must be powered on and have either the Fujitsu Custom ESXi image or a Windows/Linux OS with ServerView Agents installed.

- `restgetbootmac [<onboard|lom> [<port>]]`
  Lists the MAC address of the specified port. Default is lom 1, the ports are counted starting from 1. No plausibility checks are performed! The same requirements apply as for `restgetmac`.

- `restgetversion`
  Lists the various Rest-Profile versions. This is needed, among other things, to check the plausibility of JSON files with config data for profile management.

```
$ resgetversion
2020-08-12 18:34:26 -- Reading versions from 10.172.124.225 ...
HTTP/1,1 200 OK
Date: Wed, 12 Aug 2020 16:34:27 GMT
Server: iRMC S5 Webserver
X-Frame Options: DENY
X-XSS protection: 1; mode=block
X-Content-Type-Options: nosniff
referrer policy: no-referrer
Strict-Transport-Security: max-age=0 ; includeSubDomains
Expires: Tue, 11 Aug 2020 16:34:27 GMT
Content-Length: 407
Content-Type: application/json; charset=UTF-8

{
    "Server":{
        "@Version": "1.01",
        "AdapterConfigIrmc":{
            "@Version": "1.00"
        },
        "OSInstallation":{
            "@Version": "1.03"
        },
        "HWConfiguration":{
            "@Version": "1.02"
        },
```

```
            "HWConfigurationIrmc":{
                "@Version": "1.00"
            },
            "SystemConfig":{
                "IrmcConfig":{
                    "@Version": "1.05"
                },
                "BiosConfig":{
                    "@Version": "1.05"
                }
            }
        }
    }
```

Security concerns

Even if it is possible to enter user names and passwords via commandline parameters to all commands: This should be used only in non critical environments (e.g. for testing). Otherwise this data could be read by any other user (e.g. by `ps -ef`).

The preferred and secure way to provide those critical data to the scripts is by defining those data in either `restoptions` file, in your `~/.profile` (or `~/.bash_profile`) file or by exporting vars in your shell (e.g. `export IRMC_CRED=_admin:mysecretpassword`).

| WARNING | And of course, those files should be readable only by their owner (e.g. `chmod go-rwx restoptions ~/.profile`)! |

Bibliography

- [API] Fujitsu iRMC RESTful API
- [ProfileSpec] Profile Specification
- [GUI] iRMC S5 - Weboberfläche

All documents mentioned here (and others for the iRMC) can be found at https://support.ts.fujitsu.com (Navigation: Select a new Product/Browse for Product/Software/ServerView/Operation/Out of Band Management & iRMC/OS Independent) BIOS, Firmware, etc.)/Documents)