FortiADC™ Basic Programming REST API

October 2, 2018

FortiADC Deployment Guide: Basic Deployment Rest API

TABLE OF CONTENTS

The FortiADC REST API

Software Versions

The API described within this document is as supported by **FortiADC 5.1 and upwards.**

Introduction to REST

An API (Application Programming Interface) is a set of defined interfaces to accomplish a task, such as retrieving or modifying data. FortiADC provides a Representational State Transfer (REST) API for interaction with components of the system. Programs communicate with the REST API over HTTP, the same protocol that your web browser uses to interact with web pages. The REST API is based on interactions with a web page; data is treated like a static web page:

- Fetch data by GETing a web page
- Add data by POSTing a web page
- Update data by PUTing a web page
- Delete data by DELETEing a web page

After receiving the request, the FortiADC API sends back an HTTP response code and some payload as response data if needed.

Setting Up REST API

Unlike most other vendors, the FortiADC API is accessible without additional cost or licensing. To set up the API, just a few steps are needed.

- Give correct network settings, such as mgmt ip address and route, to make FortiADC can be accessed via network.
- Login FortiADC using **admin**, make sure the **http** and **https** have been in the **allowaccess** item of accessing interface.
- Create a new user or edit an existing one, make sure the user has **read** and **write** privilege. In the example shown, the admin account is used.
- Open a browser to load FortiADC GUI. If the FortiADC GUI can be loaded correctly, then the FortiADC REST API is ready.

Accessing the REST API

Th FortiADC REST API can be accessed from most browsers (GET). However, browser add-ons may be required for extended operations (e.g. PUT). More complicated scripted queries can be made using utilities such as cURL; most scripting languages such as Perl or Python have built in libraries for interacting with RESTful APIs. Example

shown within this document will be demonstrated with the cross platform utility cURL . All of the resource URLs are in this form:

http://[server_name]/api/[resource]

Item	Description	Example
server_name	Name or IP of the FortiADC	10.106.128.194
resource	Resource or config to be handled	load_balance_virtual_server

Resource query parameters

Queries to the API can take some parameters with resource. Below are some parameters that can be passed to the REST API URL. Please refer to the Restful API Reference documentation to find out which of parameters are allowed.

Parameter	Description	Example
?mkey=[mkey]	Member id or name of the current configuration entry	system_interface?mkey=port2
?pkey=[pkey]	Parent Id or name of the current configuration entry	load_balance_pool_child_pool_member?pkey=Pool1
?vdom=[vdom]	Vdom name of API query	system_ha/brief_status?vdom=root

Supported API Methods

Method	Operation Description	Success Response Code
GET	Retrieve a list of all resources for the endpoint	200 OK
POST	Create a new resource on the given endpoint. The data being POST-ed must follow the same format as the data returned by the GET parameter	200 OK
PUT	Update the resources for the given endpoint. Any existing items will be replaced with the new data. Data must follow the same format as the data returned by the GET parameter	200 OK
DELETE	Delete an existing resource from an endpoint	200 OK

Supported Data Formats

Currently, just JSON are supported by FortiADC REST API. So when working with FortiADC REST API, specify an Accept HTTP header with a correct mimetype is needed.

- On every resource, you can use the Accept header to indicate the format you want to communicate in as **Accept: application/json**.
- In order to POST or PUT some data to FortiADC REST API, the data must be in JSON format, so the related header need to be specified as **Content-Type: application/json**

Status Code

The response of some FortiADC REST API will contain an status code, the status code of 0 means the operation was a success. Any status code that is a non-zero integer means an error occurred. For example:

Request:

POST http://10.106.128.194/api/load_balance_real_server

{“status”:”enable”,”address”:”10.100.100.1”,”address6”:”::”,”mkey”:”real_server1”}

Response:

{“payload”:0}

Example API Calls

Login FortiADC

- The related API information:
- URL: /api/user/login,
- Method: POST
- Header: Content-Type and Accept
- Data: {"username":"admin", "password":""}

Example:

- Command:

```
curl -H "Content-Type: application/json" -H "Accept: application/json" -X POST -d '{"username":"admin","password":""}'
 -i http://10.106.128.194/api/user/login
```

- Response

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:44:20 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 366
Connection: keep-alive
Set-Cookie: last_access_time=1533721460; Path=/; Max-Age=86400; HttpOnly
x-frame-options: SAMEORIGIN

{
    "admin":"admin",
    "expire":"2018-08-07T18:10:47-07:00",
    "radius_challenge_state":"",
    "token":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQi
OjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4Lj
E5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs"
}
```

Notice: FortiADC uses both token and cookie at same time as authorization method. So after login, the Authorization and Cookie HTTP Header MUST BE set in all following REST queries. For example,
Authorization: Bearer

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pb
iI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1
pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaN
m-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs
```

```
Cookie: last_access_time=1533721460
```

Get FortiADC info

The related API information:

- URL: /api/platform/info,
- Method: GET
- Header: Accept, Authorization and Cookie

Example:

- Command:

```
curl -H "Accept: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721460" -i http://10.106.128.194/api/platform/info
```

- Response

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:45:24 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 247
Connection: keep-alive
x-frame-options: SAMEORIGIN

{
    "payload":
        {
            "hostName":"FortiADC-VM",
            "systemTime":"Wed Aug  8 02:45:24 2018",
            "systemUpTime":"35d, 7h, 4m, 21s",
            "serialNumber":"FADV080000149591",
            "firmwareVersion":"FortiADC-VM v5.1.0,build0227,180702",
            "isSystemAdmin":true,
            "isSystemWritable":true
        }
}
```

Get FortiADC full configuration file

The related API information:

- URL: /api/downloader/config,
- Method: GET
- Parameter: entire
- Header: Authorization and Cookie

Example:

- Command:

```
curl -v -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImF
kbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfa
XAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721460" -i h
ttp://10.106.128.194/api/downloader/config?entire=enable -o full_configuration.zip
```

- Response:

```
> GET /api/downloader/config?entire=enable HTTP/1.1
> Host: 10.106.128.194
> User-Agent: curl/7.55.1
> Accept: */*
> Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1
MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs
> Cookie: last_access_time=1533721460
>
< HTTP/1.1 200 OK
< Date: Wed, 08 Aug 2018 09:46:21 GMT
< Content-Type: text/plain
< Content-Length: 19086
< Connection: keep-alive
< Accept-Ranges: bytes
< Content-Disposition: attachment; filename=FortiADC-VM_20180808_0246.zip
< Last-Modified: Wed, 08 Aug 2018 09:46:21 GMT
< Set-Cookie: last_access_time=1533721580; Path=/; HttpOnly
< x-frame-options: SAMEORIGIN
<
{ [7887 bytes data]
100 19086  100 19086    0     0  19086      0  0:00:01 --:--:--  0:00:01 26693
* Connection #0 to host 10.106.128.194 left intact
```

The output file `full_configuration.zip` is the full configuration of FortiADC.

Notice: The `Set-Cookie` response header, `last_access_time` has been updated, so the following API query should use new value into `Cookie` request header, for example, `Cookie: last_access_time=1533721580.`

Upload certificate and private key files to FortiADC

The related API information:

- URL: /api/upload/certiicate_local,
- Method: POST
- Header: Authorization and Cookie
- Data: the type MUST be PKCS12 in this case, the mkey field is for name of certificate, the cert and key fields are for path of certificate and private key files and the passwd field is password of the private key file.

```
mkey=FADC_CERT
vdom=global
type=CertKey
passwd=fortinet
cert=@/root/TFTP/cert.pem
key=@/root/TFTP/cert_key.pem
```

Example:

- Command:

```
curl -v -F 'mkey=FADC_CERT' -F 'vdom=global' -F 'type=CertKey' -F 'passwd=fortinet' -F 'cert=@/root/TFTP/4132.pem' -F 'key=@/root/TFTP/4132_key.pem' -H "Authoriz
ation: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaW
QiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533
721580" http://10.106.128.194/api/upload/certificate_local
```

- Response:

```
> POST /api/upload/certificate_local HTTP/1.1
> Host: 10.106.128.194
> User-Agent: curl/7.55.1
> Accept: */*
> Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1
MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs
> Cookie: last_access_time=1533721580
> Content-Length: 2434
> Expect: 100-continue
> Content-Type: multipart/form-data; boundary=------------------------8b98e5b1baec1fd7
>
< HTTP/1.1 100 Continue
< HTTP/1.1 200 OK
< Date: Wed, 08 Aug 2018 09:47:25 GMT
< Content-Type: application/json; charset=utf-8
< Content-Length: 13
< Connection: keep-alive
< Set-Cookie: last_access_time=1533721645; Path=/; HttpOnly
< x-frame-options: SAMEORIGIN
<
* Connection #0 to host 10.106.128.194 left intact
{"payload":0}
```

Upload PKCS12 file to FortiADC

The related API information:

- URL: `/api/upload/certiicate_local,`
- Method: `POST`
- Header: `Authorization and Cookie`
- Data: the `type` MUST be `PKCS12` in this case, the `mkey` is the name of certificate, the `cert` is the path of pkcs12 file and the `passwd` is password for the file.

```
mkey=FADC_PKCS12
vdom=global
type=PKCS12
passwd=fortinet
cert=@/root/TFTP/cert_pkcs12.pfx
```

Example:

- Command:

```
curl -v -F 'mkey=FADC_PKCS12' -F 'vdom=global' -F 'type=PKCS12' -F 'passwd=fortinet' -F 'cert=@/root/TFTP/cert_pkcs12.pfx' -H "Authorization: Bearer eyJhbGciOiJI
UzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0
IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721645" http://10.106.128.
194/api/upload/certificate_local
```

- Response:

```
> POST /api/upload/certificate_local HTTP/1.1
> Host: 10.106.128.194
> User-Agent: curl/7.55.1
> Accept: */*
> Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1
MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs
> Cookie: last_access_time=1533721645
> Content-Length: 2056
> Expect: 100-continue
> Content-Type: multipart/form-data; boundary=------------------------6d28863bbd65507a
>
< HTTP/1.1 100 Continue
< HTTP/1.1 200 OK
< Date: Wed, 08 Aug 2018 09:49:02 GMT
< Content-Type: application/json; charset=utf-8
< Content-Length: 13
< Connection: keep-alive
< Set-Cookie: last_access_time=1533721742; Path=/; HttpOnly
< x-frame-options: SAMEORIGIN
<
* Connection #0 to host 10.106.128.194 left intact
{"payload":0}
```

Edit interface settings

The related API information:

URL: `/api/system_interface,`

Method: `PUT`

Parameter: `mkey`

Header: `Content-Type, Accept, Authorization and Cookie`

Data: `{"allowaccess":"https ping ssh snmp http telnet","ip":"20.3.1.10/24","ip6":"::/0","mkey":"port2"}`

Example:

- Command:

```
curl -H "Content-Type: application/json" -H "Accept: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk
1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5
OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721742" -X PUT -d '{"allowaccess":"https ping ssh snmp http telnet","ip":"20.
3.1.10/24","ip6":"::/0","mkey":"port2"}' -i http://10.106.128.194/api/system_interface?mkey=port2
```

Response:

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:50:19 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 13
Connection: keep-alive
x-frame-options: SAMEORIGIN

{"payload":0}
```

Create a real server entry

The related API information:

URL: `/api/load_balance_real_server,`

Method: `POST`

Header: `Content-Type, Accept, Authorization and Cookie`

Data: `{"status": "enable", "address": "20.3.2.1", "address6": "::", "mkey": "RealServer01"}`

Example:

- Command:

```
curl -H "Content-Type: application/json" -H "Accept: application/json" -H "Accept: application/json" -H "Accept: application/json" -H "Authorization: Bearer eyJh
bGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9y
aWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721742" -X POST -d
 '{"status": "enable", "address": "20.3.2.1", "address6": "::", "mkey": "RealServer01"}' -i http://10.106.128.194/api/load_balance_real_server
```

- Response

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:50:59 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 13
Connection: keep-alive
Set-Cookie: last_access_time=1533721859; Path=/; HttpOnly
x-frame-options: SAMEORIGIN

{"payload":0}
```

Verify the real server entry

The related API information:

URL: `/api/load_balance_real_server,`

Method: `GET`

Header: `Accept, Authorization and Cookie`

Example:

- Command:

```
curl -H "Accept: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2l
laWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwj
eyawwZooGCjs" -H "Cookie: last_access_time=1533721859" -i http://10.106.128.194/api/load_balance_real_server
```

- Response:

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:51:53 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 127
Connection: keep-alive
Set-Cookie: last_access_time=1533721913; Path=/; HttpOnly
x-frame-options: SAMEORIGIN

{
    "payload": [
        {
            "_nondeletable": 0,
            "_noneditable": 0,
            "address": "20.3.2.1",
            "address6": "::",
            "mkey": "RealServer01",
            "status": "enable"
        }
    ]
}
```

Create a pool entry

The related API information:

URL: `/api/load_balance_pool,`

Method: `POST`

Header: `Content-Type, Accept, Authorization and Cookie`

Data: `{"pool_type":"ipv4","health_check":"enable","health_check_relationship":"AND","mkey":"Pool1","health_check_list":"LB_HLTHCK_HTTP LB_HLTHCK_ICMP","rs_profile":"NONE"}`

Example:

- Command:

```
curl -H "Content-Type: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721913" -X POST -d '{"pool_type":"ipv4","health_check":"enable","health_check_relationship":"AND","mkey":"Pool1","health_check_list":"LB_HLTHCK_HTTP LB_HLTHCK_ICMP","rs_profile":"NONE"}' -i http://10.106.128.194/api/load_balance_pool
```

- Response

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:52:56 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 13
Connection: keep-alive
Set-Cookie: last_access_time=1533721976; Path=/; HttpOnly
x-frame-options: SAMEORIGIN

{"payload":0}
```

Add a real server into pool member

The related API information:

URL: `/api/load_balance_pool_child_pool_member,`

Method: `POST`

Parameter: `pkey`

Header: `Content-Type, Accept, Authorization and Cookie`

Data: `{"health_check_inherit":"enable","m_health_check":"disable","port":"80","weight":"1","connlimit":"0","recover":"0","warmup":"0","warmrate":"100","status":"enable","ssl":"disable","rs_profile_inherit":"enable","backup":"disable","cookie":"RS1","real_server_id":"RealServer01"}`

Example:

- Command:

```
curl -H "Content-Type: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533721976" -X POST -d '{"health_check_inherit":"enable","m_health_check":"disable","port":"80","weight":"1","connlimit":"0","recover":"0","warmup":"0","warmrate":"100","status":"enable","ssl":"disable","rs_profile_inherit":"enable","backup":"disable","cookie":"RS1","real_server_id":"RealServer01"}' -i http://10.106.128.194/api/load_balance_pool_child_pool_member?pkey=Pool1
```

- Response

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:53:33 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 13
Connection: keep-alive
Set-Cookie: last_access_time=1533722013; Path=/; HttpOnly
x-frame-options: SAMEORIGIN

{"payload":0}
```

Add a virtual server entry

The related API information:

URL: `/api/load_balance_virtual_server,`

Method: `POST`

Header: `Content-Type, Accept, Authorization and Cookie`

Data: `{"status":"enable","type":"l7-load-balance","addr-type":"ipv4","address":"20.3.1.100","address6":"::","port":"8088","protocol":"0","connection-limit":"1000000","content-routing":"disable","content-rewriting":"disable","error-msg":"Server-unavailable!","warmup":"0","warmrate":"100","connection-rate-limit":"0","traffic-log":"disable","trans-rate-limit":"0","scripting_flag":"disable","fortiview":"disable","mkey":"VS1","interface":"port2","profile":"LB_PROF_HTTP","persistence":"","method":"LB_METHOD_ROUND_ROBIN","pool":"Pool1","error-page":"","comments":"","traffic-group":"default","av-profile":""}`

Example:

- Command:

```
curl -H "Content-Type: application/json" -H "Accept: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk
1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2llaWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5
OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwjeyawwZooGCjs" -H "Cookie: last_access_time=1533722013" -X POST -d '{"status":"enable","type":"l7-load-balance","addr-type":"i
pv4","address":"20.3.1.100","address6":"::","port":"8088","protocol":"0","connection-limit":"1000000","content-routing":"disable","content-rewriting":"disabl
e","error-msg":"Server-unavailable!","warmup":"0","warmrate":"100","connection-rate-limit":"0","traffic-log":"disable","trans-rate-limit":"0","scripting_flag":"d
isable","fortiview":"disable","mkey":"VS1","interface":"port2","profile":"LB_PROF_HTTP","persistence":"","method":"LB_METHOD_ROUND_ROBIN","pool":"Pool1","error-p
age":"","comments":"","traffic-group":"default","av-profile":""}' -i http://10.106.128.194/api/load_balance_virtual_server
```

- Response:

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:54:13 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 13
Connection: keep-alive
Set-Cookie: last_access_time=1533722053; Path=/; HttpOnly
x-frame-options: SAMEORIGIN

{"payload":0}
```

Logout FortiADC

The related API information:

- URL: `/api/user/logout,`
- Method: `GET`
- Header: `Accept, Authorization and Cookie`

Example:

- Command:

```
curl -H "Accept: application/json" -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJTTiI6IkZBRFYwODAwMDAxNDk1OTEiLCJhZG1pbiI6ImFkbWluIiwiY29va2l
laWQiOjE5OTM0NTQ3MDAsImV4cCI6MTUzMzc1MDI2MCwiaWQiOiJhZG1pbiIsIm9yaWdfaWF0IjoxNTMzNzIxNDYwLCJyZW1vdGVfaXAiOiIxMC4xMDYuMTI4LjE5OCJ9.UaNm-19KjTJTRyuS0uDgIZprJjH0mwj
eyawwZooGCjs" -H "Cookie: last_access_time=1533722053" -i http://10.106.128.194/api/user/logout
```

- Response:

```
HTTP/1.1 200 OK
Date: Wed, 08 Aug 2018 09:54:52 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 13
Connection: keep-alive
Set-Cookie: last_access_time=1533722092; Path=/; HttpOnly
x-frame-options: SAMEORIGIN

{"payload":0}
```