Reference Guide

Revision A

McAfee Advanced Threat Defense 3.8.0

McAfee Advanced Threat Defense APIs

McAfee Advanced Threat Defense provides an Application Programming Interface (API) framework for external applications to access core McAfee Advanced Threat Defense functions through the REST protocol.

REST stands for Representational State Transfer. It relies on a stateless, client-server and cacheable communication protocol – HTTP. It is an architecture style for designing networked applications. RESTful applications use HTTP requests to post data (create and/or update), get data (query information) and delete data. Thus, REST uses HTTP for all CRUD (Create/Read/Update/Delete) operations. It is a lightweight alternative to mechanisms like RPC (Remote Procedure Calls) and Web Services such as SOAP and WSDL.

Login

This URL allows a third party application to log on to McAfee Advanced Threat Defense API framework .

Resource URL

```
GET https://<MATD_IP>/php/session.php
```

Versioning of REST API:

With this release, a new optional request parameter, `VE-API-Version` is added. As `VE-API-Version` is an optional request parameter, it does not affect the regular functionality of the client even in its absence. However, for older version 32-bit clients, Load Balancing feature is not supported. Also, at the time of McAfee Advanced Threat Defense logon, the REST API returns server API version as `apiVersion: 1.5.0`.

Input parameters

The following HTTP headers should be specified in the session request:

- Accept: application/vnd.ve.v1.0+json
- Content-Type: application/json
- VE-SDK-API: Base64 encoded "user name:password" string
- VE-API-Version (Optional)

The following two input parameters must be base64 encoded and specified in the header.

Input parameter	Description	Data type
`user name`	Logon user name.	String
`password`	Logon user password	String

All other URL resources in the ATD RESTful API are required to pass these credentials for validation and authorization in VE-SDK-API custom header.

Output parameters

Output parameter	Description	Data type
`session`	Logged on session id.	String
`userId`	Logged on user id.	String
`apiVersion`	API version	String
`matdVersion`	McAfee Advanced Threat Defense version	String

Endpoint products need to parse the following parameters.

- If the response returns "success": false, then check "reason" to know the reason.
- If the response returns "success": true, then do the following.
  - Check whether `isCurrentAPI` returns `true` or `false`
  - If `isCurrentAPI` returns `false`, then check the "warning" section to know the reason
  - If `isCurrentAPI` returns `true`, then you are on the current version

Example

When client is on latest version:

Input

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "admin:test123" string
"VE-API-Version: 1.5.0"
```

Output

```
{"success": true, "results": {"session": "ej1pgtj2u1k377qs2ejaf2dqs1","userId":
"1","isAdmin": "1","serverTZ": "PDT","apiVersion": "1.5.0","isCurrentAPI": true,
"matdVersion": "3.x.x.x.x"} }
```

When client is on a future version:

Input

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "admin:test123" string
"VE-API-Version: 1.6.0"
```

Output

```
{"success": false, "results": {"apiVersion": "1.5.0", "matdVersion": "3.x.x.x.x",
"isCurrentAPI": false, "reason": "Client API version(1.6.0) is too early, MATD supports the
Client API version till 1.5.0"}}
```

Client is on an older version which is backward compatible:

Input

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "admin:test123" string
"VE-API-Version: 1.0.0"
```

Output

```
{"success": true, "results": {"session": "ens446tbnhirtfcm7n4v6qqcs0","userId":
"1","isAdmin": "1","serverTZ": "PDT","apiVersion": "1.5.0","isCurrentAPI": false,
"matdVersion": "3.x.x.x.x","warning": "Client API version(1.0.0) is older than Server API
version(1.5.0), some feature may not be available, please refer REST API-1.5.0
documentation"} }
```

Client is on an older version which has no backward compatibility:

Input

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "admin:test123" string
"VE-API-Version: 0.7.6"
```

Output

```
{"success": false, "results": {"apiVersion": "1.5.0",  "matdVersion": "3.x.x.x.x",
"isCurrentAPI": false,  "reason": "Client API version(0.7.6) is too old, Please upgrade the
REST client version to 1.5.0"}}
```

Existing clients who cannot change their client code:

In case of no custom header (`'VE-API-Version : <API-Version>'`) available from the REST client, current version of client is considered as 1.5.0.

Input

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "admin:test123" string
```

Output

```
{"success": true, "results": {"session": "bitu3eaq2ovddl0741o84qbbj5","userId":
"1","isAdmin": "1","serverTZ": "PDT","apiVersion": "1.5.0","matdVersion": "3.x.x.x.x"} }
```

Heartbeat

This URL provides McAfee Advanced Threat Defense availability information to the user.

Resource URL

```
GET https://<MATD_IP>/php/heartbeat.php
```

Input parameters

The following HTTP headers should be specified in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- Content-Type: application/json
- VE-SDK-API: Base64 encoded "session:user id" string

Input parameters are only the following mandatory header parameters that you must supply in all calls (except File Submission). Going forward, this document does not mention these two parameters.

Input parameter	Description	Data type
session	Logged on session id.	String
userId	Logged on user id.	String

All other URL resources in the ATD RESTful API are required to pass these credentials for validation and authorization in VE-SDK-API custom header.

Output parameters

Output parameter	Description	Data type
session	Logged on session id.	String
userId	Logged on user id.	String
heartBeat	The time in seconds when the response was returned.	Numeric

Example

Input

In the HTTP header:

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "u5hiesvp9nmv9ti44vnpoi27b5:1" string
```

Output

```
{"success": true, "results": {"session": "u5hiesvp9nmv9ti44vnpoi27b5","userId":
"1","heartBeat":
"1362247723"} }
```

File/URL submission

The URL below is to upload a file/web URL for dynamic analysis by using the provided Analyzer Profile. Only single file/web URL can be submitted at a time.

Resource URL

```
POST https://<MATD_IP>/php/fileupload.php
```

The following HTTP headers should be specified in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- VE-SDK-API: Base64 encoded "session:user id" string

You can specify an optional `Expect:` parameter in the HTTP header.

You can specify an optional `skipTaskId:` (string - 1 or 0) in REST API. The performance of McAfee Advanced Threat Defense improves for version 1.5.0 and above, as REST client adds the new optional request parameter: `'skipTaskId': '1'`. If endpoint products use `'skipTaskId':'1'`, API returns the JSON response with `taskId -1` along with actual Job Id, without waiting for the `taskId` from AMAS.

Endpoint products need to use 64-bit integer instead of 32-bit for taskId.

Input parameters

Input parameter	Description
`amas_filename`	The name of the sample file. This parameter is optional for URL submissions.

`data`	Contains the following parameters defined in a json string.
	• `vmProfileList`: Analyzer profile ID. The profile ID number can be found in the UI Policy/Analyzer Profile page.
	• `submitType`: This parameter accepts four values — '0', '1', '2' and '3'.
	  • 0 — Regular file upload
	  • 1 — URL submission — URL link is processed inside analyzer VM
	  • 2 — Submit file with URL
	  • 3 — URL Download — File from URL is firstly downloaded and then analyzed
	• `url`: Any valid web URL.
	• `messageId`: (Optional) Maximum 128-character string.
	• `srcIp`: (Optional) IPv4 address of the source system or gateway from where the file is downloaded.
	• `destIp`: (Optional) IPv4 address of the target endpoint.
	• `skipTaskId`: Optional parameter with values either 0 or 1.
	Value '0' indicates corresponding taskid in API response. Value '1' indicates -1 as taskid in API response.
	• `analyzeAgain`: Optional parameter with values either 0 or 1.
	Value '0' indicates skip sample analysis if it is analyzed previously . Value '1' indicates do not skip sample analysis if it is not analyzed previously.
	• `xMode`: Optional parameter with values either 0 or 1.
	Value '0' indicates no user interaction is needed during sample analysis. Value '1' indicates user interaction is needed during sample analysis.
	• `filePriorityQ`: Optional parameter with values either `run_now` or `add_to_q`. This parameter indicates priority of sample analysis. `run_now` assigns highest priority

Input parameter	Description

	(i.e., sample is analyzed right away), `add_to_q` puts sample in waiting state if there is a waiting queue of samples.
	For `submitType` '2', we submit the file and also the URL from which the file is downloaded. McAfee GTI URL look up is done on the submitted URL in addition to file analysis.
	Examples:
	`{'data': '{"data":{"xMode":0,"overrideOS": 1,"messageId":"","vmProfileList":"12","submitType":"0","url":""}, "filePriorityQ":"run_now" }'}`
	`{'data': '{"data": {"vmProfileList":"1","messageId":"04788b1b-8dbe-4dd3-94cb-a129552af5de", "submitType":1,"url":"http://www.google.com/news"}, "filePriorityQ":"run_now" }'}`
	`{'data': '{"data": {"vmProfileList":"1","messageId":"03188b1b-8dbe-a4a3-94cb-a129552af5ee", "submitType":2,"url":"http://the.earth.li/~sgtatham/putty/latest/"}, "filePriorityQ":"add_to_q" }'}`
	`{'data': '{"data": {"vmProfileList":"11","messageId":"06488b1b-8dbe-a4c3-94cb-a129552af5dd" ,"submitType":3,"url":"http://www.javascriptenlightenment.com/ JavaScript_Enlightenment.pdf"}, "filePriorityQ":"run_now" }'}`

Output parameters

Output parameter	Description
`Results`	Contains json data with following parameters. `md5`: Hash value of the submitted sample. `subId`: JobId assigned for the sample. `taskId`: Assigned for the submitted sample. taskId is -1 for a zip file and has the same value (-1), in case `skipTaskId` is enabled. `messageId`: String that is sent in the request to identify the sample. `filesWait`: Number of samples in waiting state. `estimatedTime`: Estimated time for the analysis to finish on the submitted sample.

Example

submitType:0

Input

An example of data json string:

```
{'data': '{"data":{"xMode":0,"overrideOS":
1,"messageId":"","vmProfileList":"11","submitType":"0","url":""},
"filePriorityQ":"run_now" }'}
```

Client sends the input stream of sample to the fileupload.php. An example in Python:

```
postdata = {'data': '{"data":{"xMode":0,"overrideOS":
1,"messageId":"","vmProfileList":"11","submitType":"0","url":""},
"filePriorityQ":"run_now" }'}
```

```
file_up = {'amas_filename':open('/home/samples/temp/vtest32.exe','r')}
file_upload_req =requests.post(url,postdata,files=file_up,headers=headers,verify=False)
```

Output

```
{"success": true,"subId": 213,"mimeType": "application\/x-dosexec","filesWait":
1,"estimatedTime": 66,"results": [{"taskId": 213,"messageId": "","file":
"vtest32.exe","submitType": 0,"url": "","destIp": null,"srcIp": null,"md5":
"e2cfe1c89703352c42763e4b458fc356","size": 45056}]}
```

submitType:1

Input

An example of data json string:

```
{'data': '{"data":{"xMode":0,"overrideOS":
1,"messageId":"","vmProfileList":"11","submitType":"1","url":"http://www.yahoo.com"}}'}
```

Client sends the input stream of sample to the fileupload.php. An example in Python:

```
postdata = {'data': '{"data":{"xMode":0,"overrideOS":
1,"messageId":"","vmProfileList":"12","submitType":"1","url":"http://news.google.co.in/"}}'}
upload_rest_req = requests.post(url,postdata,headers=headers,verify=False)
```

Output

```
{"success": true,"subId": 17,"mimeType": "text\/plain","filesWait": 1,"estimatedTime":
88,"results": [{"taskId": 23,"messageId": "","file": "URL1419314922.url","submitType":
1,"url": "http: \/\/news.google.co.in\/","destIp": null,"srcIp": null,"md5":
"839f551f97e669dddb348bddb907d32c","size": 25}]}
```

submitType:2

Input

An example of data json string:

```
{'data': '{"data":
{"vmProfileList":"1","messageId":"06488b1b-8dbe-a4c3-94cb-a129552af5dd","submitType":
2,"url":"http://the.earth.li/~sgtatham/putty/latest/x86/"}’}
```

Client sends the input stream of sample to the fileupload.php. An example in Python:

```
postdata = {'data': '{"data":{"xMode":0,"overrideOS":
1,"messageId":"","vmProfileList":"12","submitType":"2","url":"http://the.earth.li/~sgtatham/
putty/latest/x86/"}}'}
file_up = {'amas_filename':open('/home/samples/vtest32.exe','r')}
upload_rest_req = requests.post(url,postdata,files=file_up,headers=headers,verify=False)
```

Output

```
{"success": true,"subId": 16,"mimeType": "application\/x-dosexec","filesWait":
1,"estimatedTime": 77,"results": [{"taskId": 22,"messageId": "","file":
"vtest32.exe","submitType": 2,"url": "http://the.earth.li/~sgtatham/putty/latest/
x86/","destIp": null,"srcIp": null,"md5": "e2cfe1c89703352c42763e4b458fc356","size":45056}]}
```

submitType:3

Input

An example of data json string:

```
{'data': '{"data":{"xMode":0,"overrideOS":
1,"messageId":"","vmProfileList":"11","submitType":"3","url":"http://10.213.248.238/
Automation/vtest32.exe"}}'}
```

Output


```
{"success": true,"subId": 210,"mimeType": "text\/plain","filesWait": 1,"estimatedTime":
68,"results": [{"taskId": -1,"file": "URL1418981249.url","md5":
"67b32fa8adaa0ae9025920c775615b96","size": "44"}]}
```


If *skip analysis* feature is enabled and a previously analyzed file has been submitted, then the API response is as follows.


```
{"success": true, "subId": 28, "mimeType": "application/x-dosexec", "filesWait": 0,
"estimatedTime": 0, "results": [ {"taskId": 28, "file": "File was previously analyzed -
(vtest32.exe )", "md5": "E2CFE1C89703352C42763E4B458FC356", "size": 45056} ]}
```


Task ID List

Resource URL below fetches the list of task id's associated with a job id.

Resource URL

```
GET https://<MATD_IP>/php/getTaskIdList.php
```

Input parameters

Input parameter	Description
jobId	Serves as an identifier for the previously submitted file.

Output parameters

Output parameter	Description
result	If a zip file with two samples is submitted, then the response contains task id's of the two samples. If a single file is submitted, then the response contains a task id of the single sample.

Example

Input

```
https:// <MATD_IP>/php/getTaskIdList.php?jobId=64
```

Output


```
{"success":true, "result":{"taskIdList":"201"}}
```


Input

```
https:// <MATD_IP>/php/getTaskIdList.php?jobId=65
```

Output


```
{"success":true, "result":{"taskIdList":"202,203"}}
```


In the above example, if the job id of a zip file is passed, then the response contains task id's of samples in the zip file.

Bulk Sample Status

The Resource URL below is to find the status of bulk number of samples in a single query.

Resource URL

```
POST https://<MATD_IP>/php/getBulkStatus.php
```

The following HTTP headers should be specified in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- VE-SDK-API: Base64 encoded "session:user id" string

Input parameters

Input parameter	Description

`data`	Contains the below parameter defined in a json string:
	• `bulkrequest`: This parameter is a json string which accepts following sub-parameters:
	  • `numRequest`: This is a numeric filed which contains the number of samples for which status is queried. The maximum value is 100.
	  • `jobIDs`: This is an array of job ID's.
	  • `taskIDs`: This parameter accepts array of task ID's.
	  jobIDs and taskIDs parameters are mutually exclusive.
	Examples:
	`{'data': '{"bulkrequest":{"numRequest":3,"jobIDs":[41,42,47]}}'}`
	`{'data': '{"bulkrequest":{"numRequest":1,"jobIDs":[150]}}'}`
	`{'data': '{"bulkrequest":{"numRequest":1,"taskIDs":[2050]}}'}`
	`{'data': '{"bulkrequest":{"numRequest":4,"taskIDs":[100,156,142,120]}}'}`

Output parameters

Output parameter	Description
`Results`	Contains json data with following parameters: `numResponse`: This is a numeric value which represents number of samples status retrieved. This is equal to numRequest in the input data. `status`:This is an array of jobIDs/taskIDs with their analysis status and score. JobID/taskID status displays following values: • 1 — accepted • 2 — waiting • 3 — analyzing • 4 — xmode • 5 — completed • 6 — cancelled • 7 — invalid • 8 — discarded

Example

Status of single jobID:

Input

```
{'data': '{"bulkrequest":{"numRequest":1,"jobIDs":[4512]}}'}
```

Output

```
{"success":true, "results":{"bulkresponse": {"numResponse":1, "status":[{"jobID":
4512,"status":5,"score":4}]}}}
```

- When a jobID is not present in the database the status and score values in the response are '-1'.
- When a jobID of a zip file is passed as input then the least status value among all the samples in the zip file at that moment is returned.

Status of multiple jobIDs:

Input

```
{'data': '{"bulkrequest":{"numRequest":3,"jobIDs":[41,30,12]}}'}
```

Output

```
{"success":true, "results":{"bulkresponse": {"numResponse":3, "status":[{"jobID":41,"status":
5,"score":5},{"jobID":30,"status":5,"score":0},{"jobID":12,"status":5,"score":5}]}}}
```

Status of single taskID:

Input

```
{'data': '{"bulkrequest":{"numRequest":1,"taskIDs":[16090]}}'}
```

Output

```
{"success":true, "results":{"bulkresponse": {"numResponse":1, "status":[{"taskID":
16090,"status":3,"score":-6}]}}}
```

- Status '3' in the above response indicates that the sample with taskID '16090' is in analyzing state.
- When a taskID is not present in the database the status and score values are '-1'.

Status of multiple taskIDs:

Input

```
{'data': '{"bulkrequest":{"numRequest":3,"taskIDs":[100,156,16109]}}'}
```

Output

```
{"success":true, "results":{"bulkresponse": {"numResponse":3, "status":[{"taskID":
100,"status":5,"score":5},{"taskID":156,"status":5,"score":4},{"taskID":16109,"status":
2,"score":-6}]}}}
```

Status '2' in the above response indicates that the sample with taskID '16109' is in waiting state.

numRequest does not match with number of jobIDs/taskIDs:

Input

```
{'data': '{"bulkrequest":{"numRequest":4,"taskIDs":[80,12,15]}}'}
```

Output

```
{"success": false, "results": {"desc": "Invalid Request"} }
```

Check Brief Status

This URL checks the analysis status.

Resource URL

```
GET https://<MATD_IP>/php/samplestatus.php
```

This URL takes iTaskId or jobId

iTaskId Parameter:

- The iTaskId must be previously returned value in the File/URL submission step.
- Retrieve the istate or status value from the response of samplestatus.php.
  - When analysis is complete, the istate=1 or 2.
  - When sample is waiting in the queue, the istate=4.
  - When sample is being analyzed, the istate=3.
  - When analysis is failed, istate=-1.
- Only when the istate=1 or 2, continue to get the results.

Input parameters

Input parameter	Description	Data type
`iTaskId`	This is the returned iTaskId value in the submission step.	Number

Output parameters

Output parameter	Description	Data type
`success`	Success is true if the request is processed successfully else false.	String
`taskid`	Task ID of the submitted sample file.	Numeric
`istate`	istate is a numeric value as explained above.	Numeric
`status`	Current status of the sample. Example: waiting / analyzing / completed.	String
`filename`	Name of the sample file.	String
`md5`	The MD5 hash value of the sample file as calculated by McAfee Advanced Threat Defense.	Hexadecimal
`vmProfile`	The internally assigned ID for the VM Profile that was used.	Numeric
`jobid`	Job ID of the submitted sample file.	Numeric
`submitTime`	Time when the sample was uploaded for analysis.	Timestamp

Output parameter	Description	Data type
summaryFiles	Summary file available. This field is valid only when istate is either 1 or 2.	
useLogs	User logs available. This field is valid only when istate is either 1 or 2.	
asmListing	Disassembly result available. This field is valid only when istate is either 1 or 2.	
PEInfo	PE information available. This field is valid only when istate is either 1 or 2.	
family	Family similarity available. This field is valid only when istate is either 1 or 2.	
vmName	VM name based on the operating system that was used for dynamic analysis.	String
vmDesc	The user-provided description for the analyzer VM that was used for dynamic analysis.	String

jobId Parameter:

1. When jobId is passed as parameter, output json response contains only one parameter 'status' and it contains below values.
   - 5 — completed
   - 3 — analyzing
   - 2 — waiting
   - -1 — failed
   - 0 — sample submitted but taskid not generated yet
2. When jobId for a zip file is passed as parameter then the status value in the json is the minimum value of status of individual samples in the zip file.
3. When jobId is passed as parameter, output json response contains parameter 'allEngineState' and it contains below values.
   - 1 — The sample is analyzed on all VMs submitted.
   - 0 — The analysis is failed or canceled in any of the VMs submitted.

Example

Input

```
https://<MATD_IP>/php/samplestatus.php?iTaskId=52
```

Output

```
{"success":true, "results":{"userid":1,"taskid":52,"istate":3,"status":"Analyzing",
"filename":"mv0107-2.exe","md5":"b01f5c6b23f5073228aa6d1e05579be4","vmProfile":"1","jobid":
55,
"submitTime":"2013-03-02 10:08:47","summaryFiles":"0","useLogs":"0",
"asmListing":"0","PEInfo":"0",
"family":"0","vmName":"Win-XP-SP3-32bit","vmDesc":"XP SP3 32-bit with all reports"} }
```

Input

```
https://<MATD_IP>/php/samplestatus.php?jobId=69
```

Output

```
{'status': 5, 'allEngineState': 1, 'severity': 5, 'success': 'true'}
```

Get report content

Use this URL to selectively download the analysis report files.

Resource URL

```
GET https://<MATD_IP>/php/showreport.php
```

Input parameters

This URL takes iType and iTaskId or jobId or md5.

The Content-Type parameter in the HTTP header is not needed in this API. Instead specify an additional Expect: in the HTTP header.

Input parameter	Description	Data type
`iTaskId`	This must be a previously returned task ID when you submitted the sample.	Numeric
`jobId`	This must be a previously returned submission ID when you submitted the sample.	Numeric
`iType`	iType can be one of the following types: • html — HTML report • txt — Text report • xml — XML report • zip — All files packaged in a single zip file • json — Same as xml but in the JSON format • ioc - Indicators of Compromise format • stix - Structured Threat Information expression. Stix generation is disabled, by default. Use **set stixreportstatus enable** to enable it. • pdf - Portable Document Format • sample - Download sample from McAfee Advanced Threat Defense	String

Output parameters

Content of the requested result file.

Example

Input

```
https://<MATD_IP>/php/showreport.php?iTaskId=10&iType=html
```

```
https://<MATD_IP>/php/showreport.php?jobId=10&iType=json
```

```
https://<MATD_IP>/php/showreport.php?md5=E2CFE1C89703352C42763E4B458FC356&iType=html
```

Output

The output is the file content of the requested result file.

Input

```
https://<MATD_IP>/php/showreport.php?iTaskId=212&iType=sample
```

Output

The output is downloaded sample 212.zip associated with taskid 212.

`iTaskId=sample` must only be used with `iTaskId` parameter.

List the analyzer profiles

This URL is to display the analyzer profiles. Only the analyzer profiles to which the user has access are displayed.

Resource URL

```
GET https://<MATD_IP>/php/vmprofiles.php
```

Output parameters

Output parameter	Description
`results`	Displays the analyzer profiles, which the user can access.

Users with admin rights can see all analyzer profiles. Other users will see analyzer profiles created by themselves and the ones assigned to them in user profile.

Example

User - admin

Output


```
{"success":true,"results":[{"vmProfileid":1,"userid":1,"imageid":10,"maxExecTime":
180,"minExecTime":5,
"recursiveAnalysis":1,"name":"winXP","vmDesc":"winXPsp3","summary":0,"userLog":0,"asm":
0,"locBlackList":0,"mfeAV":0,"reAnalysis":0,
"gtiTS":0,"gam":0,"selectedOSName":"winXPsp3","sandbox":1,"internet":1},"vmProfileid":
10,"userid":1,"imageid":11,
"maxExecTime":180,"minExecTime":5,"recusiveAnalysis":
0,"name":"win7sp1x64","vmDesc":"win7","summary":1,"userLog":0,"reAnalysis":1,
"asm":0,"locBlackList":0,"mfeAV":0,"gtiTS":0,"gam":
0,,"selectedOSName":"win7sp1x64","sandbox":1,"internet":0}]}
```


User - <non-admin user>

Output


```
{"success":true,"results":[{"vmProfileid":11,"userid":36,"imageid":10,"maxExecTime":
180,"minExecTime":5,
"recursiveAnalysis":0,"name":"winXP","vmDesc":"winXPsp3","summary":1,"userLog":1, "asm":1,
"locBlackList":0,"mfeAV":1,"reAnalysis":0,
"gtiTS":1,"gam":1, selectedOSName":"winXPsp3","sandbox":1,"internet":0}]}
```

User - nsp

Output


```
{"success":true,"results":[{"vmProfileid":10,"userid":1,"imageid":11,"maxExecTime":
180,"minExecTime":5,
"recursiveAnalysis":0,"name":"win7sp1x64","vmDesc":"win7","summary":1,"userLog":0, "asm":0,
"locBlackList":0,"mfeAV":0,"reAnalysis":1,
"gtiTS":0,"gam":0,selectedOSName":"win7sp1x64","sandbox":1,"internet":0}]}
```


`reAnalysis` parameter mentioned above is given importance in absence of `analyzeAgain` data parameter in fileupload API for end points.

`"reAnalysis":0` indicates that **Skip files if previously analyzed** is selected in **Analyzer Profile** setting.

Users List

This URL displays the user profile information present on the McAfee Advanced Threat Defense.

Resource URL

`GET https://<MATD_IP>/php/briefUserList.php`

Input parameters

Input parameter	Description
`userType`	This is the usertype associated with a user profile. For example NSP, MWG, STAND_ALONE and so on.

Output parameters

Output parameter	Description
`results`	Contains array of json data with following parameters: • idx: This is the id assigned for the user profile [Unique Identifier of the user]. • loginId: Login-id of the users. • userType: This is the input parameter passed. Example: userType=NSP. • fullName: Name associated with the user profile.

Example

Input

```
<MATD_IP>/php/briefUserList.php?userType=MEG
```

Output


```
{"success":true,"results":[{"idx":5,"loginId":"meg","userType":"MEG","fullName":"McAfee
Email Gateway "}]}
```


Input

```
https:// <MATD_IP>/php/briefUserList.php?userType=NSP
```

Output

```
{"success":true,"results":[{"idx":2,"loginId":"nsp","userType":"NSP","fullName":"NSP User"},
{"idx":7,"loginId":"vnsp","userType":"NSP","fullName":"Virtual Network Security Platform"}]}
```

Verify blacklisted and whitelisted hash values

This URL is to check if a user submitted hash value is either blacklisted or whitelisted. Only single hash value can be verified at a time.

Resource URL

`POST https://<MATD_IP>/php/atdHashLookup.php`

Input parameters

Input parameter	Description	Data type
`data`	Contains the following parameters defined in a json string. md5: Any valid 32 digit hexadecimal number Example: {'data': '{"md5":"A3CCFD0AA0B17FD23AA9FD0D84B86C05"}'}	Hexadecimal

Output parameters

Input parameter	Description
`results`	Displays input hash value and one of the below characters • 'w' — hash value is whitelisted. • 'b' — hash value is blacklisted. • '0' — hash value is not in the whitelist or blacklist. • 'Invalid input data' — invalid hash value.

Example

Input

An example of data json string:

```
{'data': '{"md5":"A3CCFD0AA0B17FD23AA9FD0D84B86C05"}'}
```

Output

```
{"success":true, "results":{"A3CCFD0AA0B17FD23AA9FD0D84B86C05":"w"}}
```

Submit YARA and behavioral rule files

Upload a custom YARA or behavioral rule file.

Resource URL

POST `https://<MATD_IP>/php/contentUpdate.php`

Specify these HTTP headers in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- VE-SDK-API: Base64 encoded "session:user id" string

Input parameters

Input key	Input value	Description
command	customYaraUpload	Indicates that the API accepts input files.
ftype	customYaraScanner	The input file is a custom YARA file.
	customBehaviorRule	The input file is a custom behavior rule file.
contentFile	In read mode, opens the input yara file on the local machine.	Indicates the path to the YARA file on local machine.

The input file must be a .yara extension.

Output parameters

Input key	Input value	Description
success	• true • false	true indicates that the YARA or behavior rule file is valid and accepted.
message	The API call success or failure message description.	

Custom behavior rule file upload example

Input

Clients send the sample input stream to the contentUpdate.php. An example in Python:

```
postdata = {"command":"customYaraUpload", "ftype":" customBehaviorRule"}
file_up = {'contentFile': open('/root/custBehavior.yara', 'r')}
```

```
file_upload_req =requests.post(url,postdata,files=file_up,headers=headers,verify=False)
```

Output

```
{"success": "true", "message":"Custom Behavioral Rules uploaded successfully."}
```

Custom YARA file upload example

Input

Client sends the sample input stream to the contentUpdate.php. An example in Python:

```
postdata = {"command":"customYaraUpload", "ftype":" customYaraScanner"}
file_up = {'contentFile': open('/root/aliceYaraFile.yara', 'r')}
```

```
file_upload_req =requests.post(url,postdata,files=file_up,headers=headers,verify=False)
```

Output

```
{"success": "true", "message":"Custom YARA Scanner Rules uploaded successfully."}
```

Enable or disable custom behavioral rules

Enable or disable the custom behavior rule settings.

Resource URL

POST `https://<MATD_IP>/php/configloader/configCreator.php`

Specify these HTTP headers in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- VE-SDK-API: Base64 encoded "session:user id" string

Input parameters

Input key	Input value	Description
command	customBehaviorRuleSetting	The API call handles the custom YARA settings.
yaraEnable	0	Disables the custom behavior rule.
	1	Enables the custom behavior rule.

Output parameters

Input key	Input value	Description
success	true	The custom behavior rule enable or disable action succeeded.

Enable custom YARA settings example

Input

An example in Python:

```
postdata = {"command":"customBehaviorRuleSetting", "yaraEnable":'1'}
file_upload_req =requests.post(url,postdata,headers=headers,verify=False)
```

```
file_upload_req =requests.post(url,postdata,files=file_up,headers=headers,verify=False)
```

Output

```
{"success": "true"}
```

Enable or disable custom YARA scanners

Enable or disable the custom YARA scanner settings.

Resource URL

POST `https://<MATD_IP>/php/configloader/configCreator.php`

Specify these HTTP headers in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- VE-SDK-API: Base64 encoded "session:user id" string

Input parameters

Input key	Input value	Description
command	customYaraScannerSetting	The API call handles the custom YARA scanner settings.
yaraEnable	0	Disables the custom YARA scanner.
	1	Enables the custom YARA scanner.
vmProfileID	Profile ID from the analyzer profile.	The analyzer profile with the enabled or disabled **Custom Yara Scanner** checkbox.

Output parameters

Input key	Input value	Description
success	true	The custom YARA scanner enable or disable action succeeded.

Enable custom YARA settings example

Input

An example in Python:

```
postdata = {"command":" customYaraScannerSetting", "yaraEnable":'1', "vmProfileID": '1'}
file_upload_req =requests.post(url,postdata,headers=headers,verify=False)
```

Output

```
{"success": "true"}
```

Logout

This URL allows logging out from McAfee Advanced Threat Defense. It generates either a response or an error message. Proper logout must be performed in order to clear the session information; else, subsequent logon is not allowed until session timeout.

Resource URL

```
DELETE https://<MATD_IP>/php/session.php
```

Input parameters

The following HTTP headers should be specified in the resource URL request:

- Accept: application/vnd.ve.v1.0+json
- Content-Type: application/json
- VE-SDK-API: Base64 encoded "session:user id" string

Input parameters are only the following mandatory header parameters that you must be supplied in all calls.

Input parameter	Description	Data type
session	Logged on session id.	String
userId	Logged on user id.	String

Output parameters

Output parameter	Description	Data type
success	Success = true / false gives the logout success status.	
results	Return value. This value is 0 if logout is successful, otherwise an error message is returned	number

Example

Input

In the HTTP header:

```
"Accept: application/vnd.ve.v1.0+json"
"Content-Type: application/json"
"VE-SDK-API:" + base64 encoded "u5hiesvp9nmv9ti44vnpoi27b5:1" string
```

Output

Sample response for correct credentials:

```
{"success": true, "results": {"return": 0} }
```

Error information

HTTP Error code	errorMessage
401	Invalid credentials.
415	Invalid accept header or content type header.