REST API Version 1 (v1)

https://campus.barracuda.com/doc/73698476/

The REST API version 1 (v1) is now succeeded with version 3. For the version 3 documentation, refer to REST API Version 3 (v3) .

Accessing the API in Version 1

This section describes how to invoke a REST API call on the Barracuda Web Application Firewall and the expected response. Both HTTP and HTTPS URI requests are supported, but in this document our examples use only HTTP URI requests for simplicity. The HTTP and HTTPS requests should include the respective port number i.e., port 8000 for HTTP and 443 for HTTPS. **Example**:

http://{IP address of BWAF}:8000/restapi/v1/{object_id}

https://{IP address of BWAF}:443/restapi/v1/{object_id}

Where:

- **http/https** – The protocol used to invoke a REST API call.
- **IP address of BWAF** – The IP address of Barracuda Web Application Firewall to access.
- **8000/443** – The port number.
- **restapi** – The name of the API.
- **v1** – The REST API version released by the Barracuda Web Application Firewall.
- **object_id** – The name of the object to be created/retrieved/modified/deleted. An object may be a Service, Security Policy, Certificate, etc.

> The object_id is specified inside curly brackets “{}” in resource URLs.

In HTTPS requests, the Barracuda Web Application Firewall uses the same certificate used for the web interface specified in the **ADVANCED > Secure Administration > SSL Certificate Configuration** section. The common name in the certificate must match the URI where you are sending the request. For example, if the URI is https://WAF1.com:443, then the common name must be WAF1.com. You must download the certificate from the **ADVANCED > Secure Administration > Private section**, and copy it to the client machine executing the REST API calls. The example below shows the login request for HTTPS, where:

- cacert – is the parameter to verify the associated certificate.
- /tmp/ssl_private_cert.pem – the path and name of the certificate.

HTTPS Login Request

Request:

curl https://WAF1.com:443/restapi/v1/login -X POST -H Content-Type:application/json -d '{"username": "admin", "password": "admin"}' --cacert /tmp/ssl_private_cert.pem

Response:

{"token":"eyJldCI6IjEzODAyMzE3NTMyIiwidXNlciI6ImFkbWluIn0=\n"}

The Barracuda Web Application Firewall REST API provides access to a number of objects through the URLs. The administrator must follow the guidelines below to invoke a REST API call on the Barracuda Web Application Firewall.

Login Request

Before accessing the Barracuda Web Application Firewall through REST API, the administrator must send a login request and generate an access token. The login request must include the username and password to generate the token. The example below shows an HTTP request. For an HTTPS request, see HTTPS Login Request.

Example:

Request:

curl http://192.168.0.1:8000/restapi/v1/login -X POST -H Content-Type:application/json -d '{"username": "admin", "password": "admin" }'

Response:

{"token":"eyJldCI6IjEzODAyMzzgzOTMyIiwidXNlciI6ImFkbWluIn0=\n"}

The generated token is embedded with the username, password and timestamp. Hence, every request made by the user should include the generated token followed by a colon (:). See the

examples below.

Example 1: Request with token

Request:

curl http://192.168.0.1:8000/restapi/v1/virtual_services -u 'eyJldCI6IjE0NDk2NWIzIiwidXNlciI6ImFkbWluIn0=\n:' -X POST -H Content-Type:application/json -d

```
'{
"name": "demo_service",
"ip_address": "99.99.107.35",
"port": "80",
"type": "HTTP",
"address_version": "ipv4",
"vsite": "default",
"group": "default"
}'
```

Response:

```
{"id":"demo_service","token":"eyJldCI6IjE0NDQ4OTQ5MTcN2U3MjNhIiwidXNlciI6ImFkbWluIn0=\
n"}
```

Logout Request

A logout request should be sent to delete the token generated for the user.

Example:

Request:

curl http://192.168.0.1:8000/restapi/v1/logout -u 'eyJldCI6IjEzODAyMzE3NTc29yzgzOTMyIiwidXNlciI6ImFkbWluIn0=\n:' -X DELETE

Response:

{"msg":"Success"}

Request Syntax

The Barracuda Web Application Firewall supports two types of JSON requests:

- Simple JSON Request
- Nested JSON Request

Simple JSON Request

In a simple JSON request, the parameters are passed in a string with a key:value pair. For example, a request for adding a service would be:

```
curl http://192.168.0.1:8000/restapi/v1/virtual_services -u
'eyJldCI6IjEzNzk2NDA4MzciidXNlciI6ImFkbWluIn0=\n:' -X POST -H Content-Type:application/json
-d

'{
"name": "demo_service_3",
"ip_address": "10.11.16.176",
"port": "80",
"type": "http",
"address_version": "ipv4",
"vsite": "demo_vsite",
"group": "demo_vsite_group"
}'
```

Nested JSON Request

In a nested JSON request the parameters with dot (.) notation such as security.mode, security.web_firewall_policy, load_balance.algorithm, load_balance.failover_method, ssl_offloading.enable_sni, ssl_offloading.enable_tls_1, etc. are passed in as a hash or a string of key:value pairs. For example, a request for updating the values of given parameters would be:

curl http://192.168.0.1:8000/restapi/v1/virtual_services/demo_service -u 'eyJldCI6IjEzNzk2NzUwNTM2IiwidXNlciI6ImFkbWluIn0=\n:' -X PUT -H Content-Type:application/json -d

```
'{
"load_balance": {
"failover_method": "ERROR",
"algorithm": "round_robin"
},
"security": {
"mode": "PASSIVE",
"web_firewall_policy": "sharepoint"
},
"ssl_offloading": {
"enable_sni": 1,
"enable_tls_1": 1
}
}'
```

For the parameters with input tables and check boxes, the values should be passed in an array. The parameters with input tables such as *cookies_exempted* in **SECURITY POLICIES > Cookie Security**, *allowed_content_types* and *allowed_methods* in **SECURITY POLICIES > URL Protection**, and the parameters with check boxes like blocked_attack_types, custom_blocked_attack_types in **SECURITY POLICIES > URL Protection** are few examples. To pass values for such parameters and check boxes is given in the example below:

curl http://192.168.0.1:8000/restapi/v1/security_policies/new_policy -u 'eyJldCI6IjE1MDE4NTY3ZDFlIiwidXNlciI6ImFkbWluIn0=\n:' -X PUT -H Content-Type:application/json -d

```
'{
"cookie_security": {
"cookie_replay_protection_type": "none",
"allow_unrecognized_cookies": "never",
"tamper_proof_mode": "encrypted"
},
"url_protection": {
"enable": "no",
"max_content_length": "0",
"max_parameters": "0",
"maximum_upload_files": "100",
"maximum_parameter_name_length": "100",
"allowed_methods": [
```

"GET",
"POST"
],
"blocked_attack_types": [
"sql_injection",
"os_command_injection",
"cross_site_scripting",
"remote_file_inclusion"
]
},
"parameter_protection": {
"enable": "yes",
"denied_metacharacters": "%00%04%1b%08%7f%23%50",
"exception_patterns": [
"sql-quote",
"unsafe-tag"
],
"file_upload_mime_types": [
"text/html",
"image/jpeg",
"image/gif"
],
"custom_blocked_attack_types": [
"cust_attack",
"cust-attack"
]
},
"cloaking": {
"return_codes_to_exempt": [
"403"
],
"filter_response_header": "yes",
"headers_to_filter": [
"Server",
"date"
]
}
}'

API Endpoint

The Barracuda Web Application Firewall REST API endpoint for making calls with the access token is:

http://<IP address of BWAF>: 8000/restapi

Combine the endpoint with the appropriate resource URL to make a call. **Example**: To create a service group under a specific vsite, the REST API call would be http://{IP address of WAF}:8000/restapi/v1/vsites/{vsite_id}/service_groups/{service_group_id } . The **Resource Description** section below provides the resource URLs for the objects that can be added/modified/retrieved/deleted.

Resource Description