

Logs

<https://campus.barracuda.com/doc/51183755/>

The Barracuda Web Application Firewall has a comprehensive logging feature to record significant events. Events related to HTTP traffic, actions of the Barracuda Web Application Firewall, and user actions are captured in logs. These log messages enable a system administrator to:

- Obtain information about the Barracuda Web Application Firewall traffic and performance.
- Analyze logs for suspicious activity.
- Troubleshoot problems.

The following types of logs are available in the Barracuda Web Application Firewall:

- Web Firewall Logs
- Access logs
- Audit logs
- System Logs
- Network Firewall Logs

For more information on logs, see [Logging, Reporting and Monitoring](#).

To Retrieve Web Firewall Logs

URL: /v1/logs/webfirewall_logs			
Method: GET			
Description: Lists all web firewall logs.			
Parameter Name	Data Type	Mandatory	Description
Input Parameters:			
parameters	Alphanumeric	Optional	Any specific parameter name that needs to be retrieved.

Example 1: Retrieving all web firewall logs

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/webfirewall_logs -u 'eyJldCI6IjE0NjQxMTg5MjgiLCJwYXNzd29yZCI6IjY0N2MxYTZIMGQwMGI5ZTdIN2ZIMDE2MmE1InNDFiYzEzliwidXNlciI6ImFkbWluIn0=\n:' -X GET
```

Response:

```
{"value":[{"ID":"154eb350fea-3a1b50","Time":"1464235003886","Client_port":53145,"Service_I
```

```
P_Port": "99.99.9.2:80", "Follow_Up_Action": 0, "Proxy_IP": "99.99.1.117", "Attack_Category": 2, "Action": 1, "Attack_Description": 119, "Attack_Detail": "GET /inex<script> HTTP/1.0", "Severity": 1, "User_Agent": "Unknown", "Query_String": "\\", "URL": "/inex<script>", "Authenticated_User": "", "Client_type": 5, "Rule_type": 0, "Country": "US", "Referer": "", "Protocol": 0, "Proxy_Port": 53145, "Useragent_Version": "-", "Client_ip": "99.99.1.117", "Rule": "global", "Host": "99.99.9.2", "metadata": {"header": [{"Action": {"1": "LOG", "3": "REDIRECT", "0": "DENY", "2": "CLOAK"}}, {"Follow_Up_Action": {"1": "Client IP Block", "0": "None", "2": "Challenge with CAPTCHA"}}, {"Severity": {"6": "Information", "4": "Warning", "1": "Alert", "3": "Error", "0": "Emergency", "7": "Debug", "2": "Critical", "5": "Notice"}}, {"Attack_Category": {"6": "XML Violations", "11": "Limits Violation", "3": "Forceful Browsing", "7": "SQL Attacks", "9": "Auth Attacks", "12": "Outbound Attacks", "2": "Protocol Violations", "8": "FILE Attacks", "1": "Session Tamper Attacks", "4": "Injection Attacks", "0": "Other Attacks", "13": "JSON Violations", "10": "DDoS Attacks", "5": "XSS Injections"}}, {"Rule_type": {"6": "Header ACL", "4": "URL Profile", "1": "URL ACL", "3": "URL Policy", "0": "Global", "7": "JSON profile", "2": "Global URL ACL", "5": "Param Profile"}}, {"Protocol": {"1": "HTTPS", "769": "TLSv1.0", "0": "HTTP", "770": "TLSv1.1", "771": "TLSv1.2", "2": "FTP", "768": "SSLv3"}}}], "token": "eyJldCI6IjE0NjU1NDQxMjkiLCJwYXNzd29yZCI6ImUxNzFIZmZhMWE5NGRmYTU1YzA1YmU3ODJjbnZjAzZjUyYUliwidXNlci6ImFkbWludn0=\n"}
```

Example 2: Retrieving web firewall logs based on a specific filter

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/webfirewall_logs -u 'eyJldCI6IjE0NjU1NDQxMjkiLCJwYXNzd29yZCI6IjE0NjU1NDQxMjkiLCJwYXNzd29yZCI6ImUxNzFIZmZhMWE5NGRmYTU1YzA1YmU3ODJjbnZjAzZjUyYUliwidXNlci6ImFkbWludn0=\n' -X GET -G -d act_taken=0
```

Response:

```
{ "value": [ { "ID": "154ebde5b7f-3a1b50", "Time": "1464246099908", "Client_port": 35656, "Service_IP_P_Port": "99.99.9.10:80", "Follow_Up_Action": 0, "Proxy_IP": "99.99.1.117", "Attack_Category": 5, "Action": 0, "Attack_Description": 158, "Attack_Detail": "type=\"cross-site-scripting\" pattern=\"script-tag\" token=\"<SCRIPT>\" Parameter=\"name\" value=\"<SCRIPT>\"", "Severity": 1, "User_Agent": "Unknown", "Query_String": "name=<SCRIPT>", "URL": "/index.html", "Authenticated_User": "", "Client_type": 5, "Rule_type": 0, "Country": "US", "Referer": "", "Protocol": 0, "Proxy_Port": 35656, "Useragent_Version": "-", "Client_ip": "99.99.1.117", "Rule": "security-policy", "Host": "99.99.9.10" }, { "ID": "154f0adeb84-3a1b50", "Time": "1464326810513", "Client_port": 51910, "Service_IP_P_Port": "99.99.9.2:80", "Follow_Up_Action": 0, "Proxy_IP": "99.99.1.117", "Attack_Category": 2, "Action": 0, "Attack_Description": 118, "Attack_Detail": "GE6T /index.html<script>>>> HTTP/1.0", "Severity": 1, "User_Agent": "Unknown", "Query_String": "\\", "URL": "/", "Authenticated_User": "", "Client_type": 5, "Rule_type": 0, "Country": "US", "Referer": "", "Protocol": 0, "Proxy_Port": 51910, "Useragent_Version": "-", "Client_ip": "99.99.1.117", "Rule": "global", "Host": "99.99.9.2" }, "metadata": {"header": [{"Action":
```

```
{
  "1": "LOG", "3": "REDIRECT", "0": "DENY", "2": "CLOAK" },
  { "Follow_Up_Action": { "1": "Client IP Block", "0": "None", "2": "Challenge with CAPTCHA" } },
  { "Severity": { "6": "Information", "4": "Warning", "1": "Alert", "3": "Error", "0": "Emergency", "7": "Debug", "2": "Critical", "5": "Notice" } },
  { "Attack_Category": { "6": "XML Violations", "11": "Limits Violation", "3": "Forceful Browsing", "7": "SQL Attacks", "9": "Auth Attacks", "12": "Outbound Attacks", "2": "Protocol Violations", "8": "FILE Attacks", "1": "Session Tamper Attacks", "4": "Injection Attacks", "0": "Other Attacks", "13": "JSON Violations", "10": "DDoS Attacks", "5": "XSS Injections" } },
  { "Rule_type": { "6": "Header ACL", "4": "URL Profile", "1": "URL ACL", "3": "URL Policy", "0": "Global", "7": "JSON profile", "2": "Global URL ACL", "5": "Param Profile" } },
  { "Protocol": { "1": "HTTPS", "769": "TLSv1.0", "0": "HTTP", "770": "TLSv1.1", "771": "TLSv1.2", "2": "FTP", "768": "SSLv3" } } },
  "token": "eyJldCI6IjE0NDUyMDM1NDAlLCJwYXNzd29yZCI6ImM5ZjJkOGE4NGUxNGYzMTk3Y2QzMGRiYTdkLnODk3Zjg1IiwidXNlci6ImFkbWluln0=" }
}
```

Example 3: Retrieving web firewall logs based on limit and offset filters

```
curl -X GET -u
'eyJldCI6IjE0NDUyMDM1NDAlLCJwYXNzd29yZCI6ImM5ZjJkOGE4NGUxNGYzMTk3Y2QzMGRiYTdkLnODk3Zjg1IiwidXNlci6ImFkbWluln0 =' 'http://<WAF-IP/PORT>/restapi/v1/logs/webfirewall_logs?limit=10&offset=25
```

Example 4: Retrieving web firewall logs based on the given interval

```
curl
http://<WAF-IP/PORT>/restapi/v1/logs/webfirewall_logs?min_time=2015-12-20T23:22:18&max_time=2015-12-21T22:20:19 -X GET -u "token:"
```

Note: The time for the filters "min_time" and "max_time" must be specified in the following format - **YYYY-MM-DDTHH-MM-SS**.

The following table lists the web firewall log parameters:

Parameter name in web interface	Parameter name to be used in the REST API command
Time	timestamp
Severity	sev_level
Action	act_taken
Follow Up Action	followup_act
Attack Description	attack_desc
Attack Category	attk_category
Client IP	client_ip
Service IP Port	serviceip:serviceport

Rule Type	rule_type
Protocol	wf_log_protocol
Proxy IP	wf_proxyip
Proxy Port	wf_proxyport
Rule	rule_id
Attack Detail	attk_detail
User Agent	wf_useragent
Authenticated User	wf_authuser
Referer	referer
Host	apslog_host
URL	url
Useragent Version	useragent_version
Country	country_code
ID	log_uid
Query String	query_str
Client Type	client_type
Limit	limit
Offset	offset
Minimum Time	min_time
Maximum Time	max_time

To Retrieve Access Logs

URL: /v1/logs/access_logs			
Method: GET			
Description: Lists all access logs.			
Parameter Name	Data Type	Mandatory	Description
Input Parameters:			
parameters	Alphanumeric	Optional	Any specific parameter name that needs to be retrieved.

Example 1: Retrieving all access logs

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/access_logs -u
'eyJldCI6IjE0NjU1NDQzNjEiLCJwYXNzd29yZCI6Ijc4NmVhZDZIMWQ1NGVkdDQzZWE3YTU0Y2Iz\nN
WQzYjNlIiwidXNlci6ImFkbWludn0=\n:' -X GET
```



```

verIP_Port":"10.11.25.117:80","Custom_Header3":"","-
\","Proxy_IP":"99.99.1.117","Server_Time":0,"Custom_Header1":"","-
\","Time_Taken":26,"Client_Port":51910,"Authenticated_User":"","Referrer":"","-
\","Bytes_Received":38,"Profile_Matched":1,"Country":"US","Session_ID":"","Protected":2,"Client_IP":"99.99.1.117","Client_Type":5,"Encrypted_URL":"","-
\","Proxy_Port":51910,"Protocol":0,"Cookie":"","-"}, {"Web_Firewall_Matched":0,"Login":"","-
\","Response_Type":1,"Bytes_Sent":399,"Clickjacking":0,"User_Agent":"Unknown","Query_String":"","URL":"/SDGF/1'OR'1","Method":"GET","Version":"HTTP/1.0","Certificate_User":"","-
\","Custom_Header2":"","Host":"99.99.9.3","ID":"154f0b03e72-3a1b50","Time":"1464326963208","Cached":0,"ServerIP_Port":"10.11.25.117:80","Custom_Header3":"","Proxy_IP":"99.99.1.117","Server_Time":2,"Custom_Header1":"","Time_Taken":406,"Client_Port":32950,"Authenticated_User":"","Referrer":"","-
\","Bytes_Received":27,"Profile_Matched":1,"Country":"US","Session_ID":"","Protected":1,"Client_IP":"99.99.1.117","Client_Type":5,"Encrypted_URL":"","-
\","Proxy_Port":32950,"Protocol":771,"Cookie":"","-
\"}], "metadata": {"header": [{"Protected": {"1": "Passive", "0": "Unprotected", "2": "Protected"}}, {"Web_Firewall_Matched": {"1": "Invalid", "0": "Valid"}}, {"Profile_Matched": {"1": "Default", "0": "Profiled"}}, {"Response_Type": {"1": "Server", "0": "Internal"}}, {"Protocol": {"3": "WS", "770": "TLSv1.1", "771": "TLSv1.2", "2": "FTP", "1": "HTTPS", "4": "WSS", "0": "HTTP", "769": "TLSv1.0", "768": "SSLv3"}}}], "token": "eyJldCI6IjE0NDY5MTYiLCJwYXNzd29yZCI6ImU2ZmJjZjM0YWVkfODM4Y2E2NTRiNWYzZjAx\nOTg4ZDEzZWZlZmFkbWVud0="}

```

Example 3: Retrieving access logs based on limit and offset filters

```

curl -X GET --header 'Accept: application/json' -u
'eyJldCI6IjE0NDY5MTYiLCJwYXNzd29yZCI6ImU2ZmJjZjM0YWVkfODM4Y2E2NTRiNWYzZjAx\nOTg4ZDEzZWZlZmFkbWVud0 =' 'http://<WAF-IP/PORT>/restapi/v1/logs/access_logs?limit=10&offset=25

```

Example 4: Retrieving access logs based on the given interval

```

curl
http://<WAF-IP/PORT>/restapi/v1/logs/access_logs?min_time=2015-12-20T23:22:18&max_time=2015-12-21T22:20:19 -X GET -u "token:"

```

Note: The time for the filters "min_time" and "max_time" must be specified in the following format - **YYYY-MM-DDTHH-MM-SS**.

The following table lists the access log parameters:

Parameter name in web interface	Parameter name to be used in the REST API command
---------------------------------	---

Time	timestamp
ID	log_uid
Client IP	client_ip
Client Port	client_port
Country	country_code
Client Type	client_type
Certificate User	cert_user
Proxy IP	web_proxyip
Proxy Port	web_proxyport
User Agent	web_useragent
Authenticated User	web_authuser
Custom Header1	web_cusheader1
Custom Header2	web_cusheader2
Custom Header3	web_cusheader3
ServerIP Port	serverip:serverport
Method	method
Clickjacking	click_jacking
Encrypted URL	encrypted_url
Cached	cache_hit
Bytes Sent	byte_sent
Bytes Received	byte_recvd
Protected	protected_flag
Web Firewall Matched	wf_match_flag
Profile Matched	profile_flag
Response Type	response_flag
Protocol	web_log_protocol
Version	weblog_version
Host	weblog_host
URL	uri_stem
Query String	query_str
Referrer	referrer
Time Taken	time_taken
Server Time	server_time
Session ID	session_id
Limit	limit
Offset	offset

Minimum Time	min_time
Maximum Time	max_time

To Retrieve Audit Logs

URL: /v1/logs/audit_logs			
Method: GET			
Description: Lists all audit logs.			
Parameter Name	Data Type	Mandatory	Description
Input Parameters:			
parameters	Alphanumeric	Optional	Any specific parameter name that needs to be retrieved.

Example 1: Retrieving all audit logs

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/audit_logs -u
'eyJldCI6IjE0NDQzNjEiLCJwYXNzd29yZCI6Ijc4NmVhZDZIMWQ1NGVkZDQzZWE3YTU0Y2Iz\nN
WQzYjNlIiwidXNlci6ImFkbWlud0=\n:' -X GET
```

Response:

```
{
  "value": [
    {
      "ID": "56b9a08ed8ebf6113b65e895",
      "Time": "1455005838537",
      "Role": "admin",
      "Object_Name": "Data path",
      "Transaction_Type": "Initialization",
      "Additional_Data": "[Service Initialization]",
      "Transaction_ID": 0,
      "Login_IP": "127.0.0.1",
      "Object_Type": "Services",
      "Old_Value": "",
      "New_Value": "",
      "Variable": "",
      "Admin": "admin",
      "Change_Type": "Start"
    },
    {
      "ID": "56b9cafcfc5891108b6a25a6",
      "Time": "1455016699565",
      "Role": "admin",
      "Object_Name": "service_2",
      "Transaction_Type": "Config",
      "Additional_Data": "",
      "Transaction_ID": 10,
      "Login_IP": "10.11.18.25",
      "Object_Type": "service",
      "Old_Value": "",
      "New_Value": "X-Forwarded-For",
      "Variable": "aps_req_rewrite_header",
      "Admin": "admin",
      "Change_Type": "Set"
    },
    {
      "ID": "56b9cafcfc5891108b6a25a7",
      "Time": "1455016699565",
      "Role": "admin",
      "Object_Name": "service_2",
      "Transaction_Type": "Config",
      "Additional_Data": "",
      "Transaction_ID": 10,
      "Login_IP": "10.11.28.232",
      "Object_Type": "service",
      "Old_Value": "",
      "New_Value": "1455016692",
      "Variable": "service_creation_time",
      "Admin": "admin",
      "Change_Type": "Set"
    },
    {
      "ID": "56b9cafcfc5891108b6a25a8",
      "Time": "1455016699565",
      "Role": "admin",
      "Object_Name": "service_2",
      "Transaction_Type": "Config",
      "Additional_Data": "",
      "Transaction_ID": 10,
      "Login_IP": "10.11.19.89",
      "Object_Type": "service",
      "Old_Value": "",
      "New_Value": "255.255.255.255",
      "Variable": "if_mask",
      "Admin": "admin",
      "Change_Type": "Set"
    },
    {
      "ID": "56b9cafcfc5891108b6a25a9",
      "Time": "1455016699566",
      "Role": "admin",
      "Object_Name": "service_2:default-url-policy",
      "Transaction_Type": "Config",
      "Additional_Data": "",
      "Transaction_ID": 10,
      "Login_IP": "10.11.18.25",
      "Object_Type": "aps_url_acl",
      "Old_Value": "",
      "New_Value": "",
      "Variable": "",
      "Admin": "admin",
      "Change_Type": "Add"
    }
  ],
  "metadata": {
    "header": {
      "Transaction_Type": {
        "11": "Support Tunnel"
      }
    }
  }
}
```

```
closed","21":"Account Locked","7":"Shutdown","17":"Clear Statistics and
Logs","2":"Config","22":"sendgarp_executed","1":"Logout","18":"Initialization","0":"Login","23":"
failover_executed","16":"Admin Access Violation","13":"Firmware
Revert","25":"config_sync","6":"Reboot","3":"Command","9":"Energize Updates","12":"Firmware
Apply","20":"Delete Cloud Node","14":"Session-Timeout","15":"Unsuccessful
Login","8":"Firmware Update","4":"Rollback","24":"failback_executed","19":"Add Cloud
Node","10":"Support Tunnel
open","5":"Restore"}},"Change_Type":{"6":"Copy","11":"Done","3":"Delete","7":"Success","9":
"Start","2":"Modify","8":"Failure","1":"Add","4":"Set","0":"None","10":"Stop","5":"Clear"}}},"tok
en":"eyJldCI6IjE0NjU1NDY5ODYiLCJwYXNzd29yZCI6ImlyNTE2ZDIyM2VkOTI5NWJiZWZhYjZDc4\
nZjI1MzA4IiwidXNlciI6ImFkbWluln0=\n"}
```

Example 2: Retrieving audit logs based on a specific filter

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/audit_logs -u
'eyJldCI6IjE0NjU1NDY5ODYiLCJwYXNzd29yZCI6IjY0N2MxYTZIMGQwMGI5ZTdIN2ZIMDE2MmE1\nN
DFiYzEzIiwidXNlciI6ImFkbWluln0=\n' -X GET -G -d login_ip!=10.11.18.25
```

Response:

```
{ "value": [ { "ID": "56b9a08ed8ebf6113b65e895", "Time": "1455005838537", "Role": "admin", "Objec
t_Name": "Data path", "Transaction_Type": "Initialization", "Additional_Data": [ Service
Initialization ], "Transaction_ID": 0, "Login_IP": "127.0.0.1", "Object_Type": "Services", "Old_Value": "",
"New_Value": "", "Variable": "", "Admin": "admin", "Change_Type": "Start" }, { "ID": "56b9a09ed8ebf6
113b65e8a1", "Time": "1455005854253", "Role": "admin", "Object_Name": "Data
path", "Transaction_Type": "Initialization", "Additional_Data": [ Data path successfully
initialized ], "Transaction_ID": 0, "Login_IP": "127.0.0.1", "Object_Type": "Services", "Old_Value": "",
"New_Value": "", "Variable": "", "Admin": "admin", "Change_Type": "Success" }, { "ID": "56b9c0acfc589
1108b6a2575", "Time": "1455014060250", "Role": "admin", "Object_Name": "Data
path", "Transaction_Type": "Initialization", "Additional_Data": [ Service
Initialization ], "Transaction_ID": 0, "Login_IP": "127.0.0.1", "Object_Type": "Services", "Old_Value": "",
"New_Value": "", "Variable": "", "Admin": "admin", "Change_Type": "Start" }, { "metadata": { "header": [
{ "Transaction_Type": { "11": "Support Tunnel closed", "21": "Account
Locked", "7": "Shutdown", "17": "Clear Statistics and
Logs", "2": "Config", "22": "sendgarp_executed", "1": "Logout", "18": "Initialization", "0": "Login", "23": "
failover_executed", "16": "Admin Access Violation", "13": "Firmware
Revert", "25": "config_sync", "6": "Reboot", "3": "Command", "9": "Energize Updates", "12": "Firmware
Apply", "20": "Delete Cloud Node", "14": "Session-Timeout", "15": "Unsuccessful
Login", "8": "Firmware Update", "4": "Rollback", "24": "failback_executed", "19": "Add Cloud
Node", "10": "Support Tunnel
open", "5": "Restore" } }, { "Change_Type": { "6": "Copy", "11": "Done", "3": "Delete", "7": "Success", "9":
"Start", "2": "Modify", "8": "Failure", "1": "Add", "4": "Set", "0": "None", "10": "Stop", "5": "Clear" } } ] }, "tok
```

```
en": "eyJldCI6IjE0NjU1NDcxODYiLCJwYXNzd29yZCI6IjdlZGEwMjhiMzk3OGNhOGU3ZWE4MTAzOGUx\nZmRjOWEzliwidXNlcil6ImFkbWluln0=\n"} }
```

Example 3: Retrieving audit logs based on limit and offset filters

```
curl -X GET --header 'Accept: application/json' -u  
'eyJldCI6IjE1MDUyMDM1NDAlLCJwYXNzd29yZCI6ImM5ZjJkOGE4NGUxNGYzMtk3Y2QzMGRiYTdk\  
nODk3Zjg1IiwidXNlcil6ImFkbWluln0 =:' 'http://<WAF-  
IP/PORT>/restapi/v1/logs/audit_logs?limit=10&offset=25
```

Example 4: Retrieving audit logs based on the given interval

```
curl  
http://<WAF-IP/PORT>/restapi/v1/logs/audit_logs?min_time=2015-12-20T23:22:18&max_time=  
2015-12-21T22:20:19 -X GET -u "token:"
```

Note: The time for the filters "min_time" and "max_time" must be specified in the following format - **YYYY-MM-DDTHH-MM-SS**.

The following table lists the audit log parameters:

Parameter name in web interface	Parameter name to be used in the REST API command
Time	timestamp
ID	bson_oid
Login IP	login_ip
Admin	admin_name
Role	admin_role
Transaction Type	txn_name
Change Type	chg_name
Transaction ID	txn_id
Object_Type	obj_type
Object_Name	obj_name
Variable	variable
Old Value	old_value
New Value	new_value
Additional Data	add_data
Limit	limit

Offset	offset
Minimum Time	min_time
Maximum Time	max_time

To Retrieve System Logs

URL: /v1/logs/system_logs			
Method: GET			
Description: Lists all system logs.			
Parameter Name	Data Type	Mandatory	Description
Input Parameters:			
parameters	Alphanumeric	Optional	Any specific parameter name that needs to be retrieved.

Example 1: Retrieving all system logs

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/system_logs -u
'eyJldCI6IjE0NjU1NDQzNjEiLCJwYXNzd29yZCI6Ijc4NmVhZDZIMWQ1NGVkbZDQzZWE3YTU0Y2Iz\nN
WQzYjNlIiwidXNlci6ImFkbWluln0=\n:' -X GET
```

Response:

```
{ "value": [ { "ID": "56f76bfc4d1495115204049a", "Time": "1459055612510", "Event_ID": 7005, "Mes
sage": "[ALERT:7005] Server 10.11.25.117:80 is enabled by out of band monitor. Reason:out of
band
monitor", "Module": "LB", "Severity": 1 }, { "ID": "56f76bfc4d1495115204049b", "Time": "1459055612
589", "Event_ID": 56003, "Message": "Server:10.11.25.117:80 Host:- is up Reason:out of band
monitor\n", "Module": "HEALTH", "Severity": 6 }, { "ID": "56f76c0e4d1495115204049c", "Time": "1459
055630993", "Event_ID": 44047, "Message": "Memory Usage exceeds 85%.Current RAM
Usage:57%, Swap Usage:
88%", "Module": "PROCMON", "Severity": 1 }, { "ID": "56f76c4d4d1495115204049d", "Time": "145905
5693774", "Event_ID": 44047, "Message": "Memory Usage exceeds 85%.Current RAM Usage:57%,
Swap Usage:
88%", "Module": "PROCMON", "Severity": 1 }, { "ID": "56f76c8c4d1495115204049e", "Time": "145905
5756504", "Event_ID": 44047, "Message": "Memory Usage exceeds 85%.Current RAM Usage:57%,
Swap Usage:
88%", "Module": "PROCMON", "Severity": 1 }, { "ID": "56f76ccb4d1495115204049f", "Time": "145905
5819256", "Event_ID": 44047, "Message": "Memory Usage exceeds 85%.Current RAM Usage:57%,
Swap Usage:
88%", "Module": "PROCMON", "Severity": 1 }, { "ID": "56f76d0a4d149511520404a0", "Time": "145905
5882054", "Event_ID": 44047, "Message": "Memory Usage exceeds 85%.Current RAM Usage:58%,
```

Swap Usage:

```
88%,"Module":"PROCMON","Severity":1},{ "ID":"56f76f8a4d149511520404af","Time":"1459056
522819","Event_ID":7006,"Message":"[ALERT:7006] Server 10.11.25.117:80 is disabled by out
of band monitor. Reason: TCP connection timedout error
.", "Module":"LB","Severity":1},{ "ID":"56f76f8a4d149511520404b0","Time":"1459056522928","
Event_ID":56004,"Message":"Server:10.11.25.117:80 Host:- is down Reason: TCP connection
timedout error
.\n", "Module":"HEALTH","Severity":1},{ "ID":"56f76f944d149511520404b1","Time":"145905653
2821","Event_ID":7005,"Message":"[ALERT:7005] Server 10.11.25.117:80 is enabled by out of
band monitor. Reason:out of band
monitor", "Module":"LB","Severity":1},{ "ID":"56f76f944d149511520404b2","Time":"1459056532
902","Event_ID":56003,"Message":"Server:10.11.25.117:80 Host:- is up Reason:out of band
monitor\n", "Module":"HEALTH","Severity":6}], "metadata": { "header": { "Severity": { "6": "6-
Information", "4": "4-Warning", "1": "1-Alert", "3": "3-Error", "0": "0-Emergency", "7": "7-
Debug", "2": "2-Critical", "5": "5-
Notice" } } }, "token": "eyJldCI6IjE0NjU1NDc0MDUjLCJwYXNzd29yZCI6ImVhOWE1ZjlyNDM3MmI0N
TM4NTg4MDAyZmY2\nYWQwMDhiliwidXNlci6ImFkbWluln0=\n"} }
```

Example 2: Retrieving system logs based on a specific filter**Request:**

```
curl http://10.11.25.9:8000/restapi/v1/logs/system_logs -u
'eyJldCI6IjE0NjU1NDQzNjEiLCJwYXNzd29yZCI6Ijc4NmVhZDZIMWQ1NGVhZDQzZWE3YTU0Y2Iz\nN
WQzYjNlliwidXNlci6ImFkbWluln0=\n' -X GET -G -d module_name=LB
```

Response:

```
{ "value": [ { "ID": "56f76bfc4d1495115204049a", "Time": "1459055612510", "Event_ID": 7005, "Mes
sage": "[ALERT:7005] Server 10.11.25.117:80 is enabled by out of band
band
monitor", "Module": "LB", "Severity": 1 }, { "ID": "56f76db54d149511520404a3", "Time": "145905605
3007", "Event_ID": 7006, "Message": "[ALERT:7006] Server 10.11.25.117:80 is disabled by out of
band monitor. Reason: TCP connection timedout error
.", "Module": "LB", "Severity": 1 }, { "ID": "56f76dc54d149511520404a5", "Time": "1459056069534", "
Event_ID": 7005, "Message": "[ALERT:7005] Server 10.11.25.117:80 is enabled by out of band
monitor. Reason:out of band
monitor", "Module": "LB", "Severity": 1 }, { "ID": "56f76f8a4d149511520404af", "Time": "1459056522
819", "Event_ID": 7006, "Message": "[ALERT:7006] Server 10.11.25.117:80 is disabled by out of
band monitor. Reason: TCP connection timedout error
.", "Module": "LB", "Severity": 1 }, { "ID": "56f76f944d149511520404b1", "Time": "1459056532821", "
Event_ID": 7005, "Message": "[ALERT:7005] Server 10.11.25.117:80 is enabled by out of band
monitor. Reason:out of band
monitor", "Module": "LB", "Severity": 1 }, { "ID": "56f777a34d149511520404f8", "Time": "1459058595
```

```
430,"Event_ID":7006,"Message":"[ALERT:7006] Server 10.11.25.117:80 is disabled by out of
band monitor. Reason: TCP connection timedout error
.", "Module": "LB", "Severity": 1}, {"ID": "56f777ad4d149511520404fa", "Time": "1459058605432", "
Event_ID": 7005, "Message": "[ALERT:7005] Server 10.11.25.117:80 is enabled by out of band
monitor. Reason: out of band
monitor", "Module": "LB", "Severity": 1}], "metadata": {"header": [{"Severity": {"6": "6-
Information", "4": "4-Warning", "1": "1-Alert", "3": "3-Error", "0": "0-Emergency", "7": "7-
Debug", "2": "2-Critical", "5": "5-
Notice"} } ]}, "token": "eyJldCI6IjE0NjU1NDc0NTEiLCJwYXNzd29yZCI6ImE1MmFhNmRiNGRmNDhm
Yzg2YmJhMzdiNGYz\nZTYyYzliwidXNlci6ImFkbWluln0=\n"} }
```

Example 3: Retrieving system logs based on limit and offset filters

```
curl -X GET --header 'Accept: application/json' -u
'eyJldCI6IjE1MDUyMDM1NDAlLCJwYXNzd29yZCI6ImM5ZjJkOGE4NGUxNGYzMTk3Y2QzMGRiYTdk\
nODk3Zjg1liwidXNlci6ImFkbWluln0 =' 'http://<WAF-
IP/PORT>/restapi/v1/logs/system_logs?limit=10&offset=25
```

Example 4: Retrieving system logs based on the given interval

```
curl
http://<WAF-IP/PORT>/restapi/v1/logs/system_logs?min_time=2015-12-20T23:22:18&max_time
=2015-12-21T22:20:19 -X GET -u "token:"
```

Note: The time for the filters "min_time" and "max_time" must be specified in the following format - **YYYY-MM-DDTHH-MM-SS**.

The following table lists the system log parameters:

Parameter name in web interface	Parameter name to be used in the REST API command
Time	timestamp
Module	module_name
ID	bson_oid
Event ID	event_id
Severity	sev_level
Message	log_msg
Limit	limit
Offset	offset
Minimum Time	min_time

Maximum Time	max_time
--------------	----------

To Retrieve Network Firewall Logs

URL: /v1/logs/			
Method: GET			
Description: Lists all network firewall logs.			
Parameter Name	Data Type	Mandatory	Description
Input Parameters:			
parameters	Alphanumeric	Optional	Any specific parameter name that needs to be retrieved.

Example 1: Retrieving all network firewall logs

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/nwfirewall_logs -u
'eyJldCI6IjE0NjU1NDQzNjEiLCJwYXNzd29yZCI6IjY4NmVhZDZIMWQ1NGVkZDQzZWE3YTU0Y2Iz\nN
WQzYjNlliwidXNlciI6ImFkbWluln0=\n' -X GET
```

Response:

```
{
  "value": [
    {
      "ID": "5718af7a4d149511670ffd7a",
      "Source_Port": 29926,
      "Time": "1461235578777",
      "Destination_Port": 80,
      "Destination_IP": "99.99.9.101",
      "Source_IP": "1.169.193.215",
      "ACL_Policy": 0,
      "Country": "TW",
      "Protocol": "TCP",
      "ACL_Name": "GeolIP-Pool:abc"
    },
    {
      "ID": "5718b2664d149511670ffd7b",
      "Source_Port": 60625,
      "Time": "1461236326053",
      "Destination_Port": 80,
      "Destination_IP": "99.99.9.101",
      "Source_IP": "103.240.91.7",
      "ACL_Policy": 0,
      "Country": "IN",
      "Protocol": "TCP",
      "ACL_Name": "TOR-Nodes"
    },
    {
      "ID": "5718b3e44d149511670ffd7e",
      "Source_Port": 30694,
      "Time": "1461236708320",
      "Destination_Port": 80,
      "Destination_IP": "99.99.9.101",
      "Source_IP": "1.169.193.215",
      "ACL_Policy": 0,
      "Country": "TW",
      "Protocol": "TCP",
      "ACL_Name": "GeolIP-Pool:abc"
    },
    {
      "ID": "5718b7674d149511670ffd81",
      "Source_Port": 27362,
      "Time": "1461237607188",
      "Destination_Port": 80,
      "Destination_IP": "99.99.9.101",
      "Source_IP": "1.1.160.247",
      "ACL_Policy": 0,
      "Country": "TH",
      "Protocol": "TCP",
      "ACL_Name": "Anonymous-Proxy-or-Satellite-Provider"
    }
  ],
  "metadata": {
    "header": {
      "ACL_Policy": {
        "1": "Allow",
        "0": "Deny"
      }
    }
  },
  "token": "eyJldCI6IjE0NjU1NDc1MDAiLCJwYXNzd29yZCI6IjMyOTUzM2E5ZGUwZWlzMWE1YzRjNWUzNGYz\nZTRhNG
U3liwidXNlciI6ImFkbWluln0=\n"
}
```

Example 2: Retrieving network firewall logs based on a specific filter

Request:

```
curl http://10.11.25.9:8000/restapi/v1/logs/nwfirewall_logs -u
'eyJldCI6IjE0NjQxMTg5MjgiLCJwYXNzd29yZCI6IjY0N2MxYTZlMGQwMGI5ZTdIN2ZIMDE2MmE1\nN
DFiYzEzliwidXNlciI6ImFkbWluln0=\n:' -X GET-G -d acl_id=GeolP-Pool:hello
```

Response:

```
{ "value": [ { "ID": "5718af7a4d149511670ffd7a", "Source_Port": 18826, "Time": "1461235578777", "
Destination_Port": 80, "Destination_IP": "99.99.9.101", "Source_IP": "1.169.193.215", "ACL_Policy": 0
, "Country": "TW", "Protocol": "TCP", "ACL_Name": "GeolP-
Pool:hello" }, { "ID": "5718b2664d149511670ffd7b", "Source_Port": 60625, "Time": "1461236326053
", "Destination_Port": 80, "Destination_IP": "99.99.9.101", "Source_IP": "103.240.91.7", "ACL_Policy":
0, "Country": "IN", "Protocol": "TCP", "ACL_Name": "GeolP-
Pool:hello" }, { "ID": "5718b3e44d149511670ffd7e", "Source_Port": 30694, "Time": "1461236708320
", "Destination_Port": 80, "Destination_IP": "99.99.9.101", "Source_IP": "1.169.193.215", "ACL_Policy"
: 0, "Country": "TW", "Protocol": "TCP", "ACL_Name": "GeolP-
Pool:hello" }, { "metadata": { "header": { "ACL_Policy": { "1": "Allow", "0": "Deny" } } } }, "token": "eyJldCI
6IjE0NjU1NDc1MDAiLCJwYXNzd29yZCI6IjMyOTUzM2E5ZGUwZWlzMWE1YzRjNWUzNGYz\nZTRhN
GU3liwidXNlciI6ImFkbWluln0=\n" }
```

Example 3: Retrieving network firewall logs based on limit and offset filters

```
curl -X GET --header 'Accept: application/json' -u
'eyJldCI6IjE1MDUyMDM1NDAlLCJwYXNzd29yZCI6ImM5ZjJkOGE4NGUxNGYzMTk3Y2QzMGRiYTdk\
nODk3Zjg1liwidXNlciI6ImFkbWluln0 =' 'http://<WAF-
IP/PORT>/restapi/v1/logs/nwfirewall_logs?limit=10&offset=25
```

Example 4: Retrieving network firewall logs based on the given interval

```
curl
http://<WAF-IP/PORT>/restapi/v1/logs/nwfirewall_logs?min_time=2015-12-20T23:22:18&max_ti
me=2015-12-21T22:20:19 -X GET -u "token:"
```

Note: The time for the filters "min_time" and "max_time" must be specified in the following format - **YYYY-MM-DDTHH-MM-SS**.

The following table lists the network firewall log parameters:

Parameter name in web interface	Parameter name to be used in the REST API command
Time	timestamp
ACL Name	acl_id

Source IP	src_ip
Source Port	src_port
Country	country_code
Destination IP	dest_ip
Destination Port	dest_port
ID	bson_oid
Protocol	acl_protocol
ACL Policy	acl_action
Limit	limit
Offset	offset
Minimum Time	min_time
Maximum Time	max_time

© Barracuda Networks Inc., 2024 The information contained within this document is confidential and proprietary to Barracuda Networks Inc. No portion of this document may be copied, distributed, publicized or used for other than internal documentary purposes without the written consent of an official representative of Barracuda Networks Inc. All specifications are subject to change without notice. Barracuda Networks Inc. assumes no responsibility for any inaccuracies in this document. Barracuda Networks Inc. reserves the right to change, modify, transfer, or otherwise revise this publication without notice.