

Automation API Guide

<https://campus.barracuda.com/doc/3539040/>

This article applies to:

- Barracuda Load Balancer running firmware version 3.6.1.009 or higher
- Barracuda Load Balancer 440 and above

Use the Barracuda Load Balancer API to automate a variety of tasks including:

- Add or delete a Service
- Add or delete a Real Server
- Show the state and configuration data for a Service or Real Server
- Enable or disable a Real Server or put a Real Server into maintenance mode

The interface is implemented using a set of XML-RPC requests and parameters that specify the inputs. The samples included in this article are written in Perl, but you can call the API from any programming language that supports XML-RPC calls.

See the [Sample Program](#) for a complete example Perl program.

APIs Available for the Barracuda Load Balancer

- [service.add](#) - This method adds a Service to the configuration database. The VIP address and port combination must be unique.
- [service.delete](#) - This method deletes an existing Service from the configuration database.
- [service.show](#) - This method displays configuration and state information for a Service and/or a Real Server. If no parameters are entered to identify one Service or Real Server, information about all existing Services and Real Servers is returned.
- [server.add](#) - This method adds a Real Server to the configuration database. It must be added to an existing Service.
- [server.delete](#) - This method removes an existing Real Server from an associated Service.
- [server.change_state](#) - This method changes the state of a Real Server to disabled, maintenance or enabled. Changing it to disabled terminates all connections immediately. When placed in maintenance mode, the Real Server keeps existing connections but does not accept any new ones. Maintenance can be performed when all active connections are closed.

Configuring the Barracuda Load Balancer to Accept API Calls

Before running any program, configure the Barracuda Load Balancer to accept calls from the external

system.

On the Barracuda Load Balancer,

1. Navigate to the **BASIC > Administration** page.
2. In the **Allowed IP/Range** table, enter the IP address of the system from where the API call is to be made.
3. Enter an API password.

Attempts to use the API from an IP address that is not in the table are denied.

Installing Perl and XML-RPC on the Control System

Because the APIs are all XML-RPC methods, XML-RPC must be installed on the control system.

To install perl and XML-RPC,

Linux systems

1. Install perl.
2. To start the CPAN shell, type:
`perl -MCPAN -we 'shell'`
3. From the CPAN shell, type:
`install XML::RPC`

Windows systems

1. Install perl (for example, ActivePerl or Strawberry Perl).
2. From the command prompt, type:
`cpan.bat`
3. Once in CPAN, type:
`install XML::RPC`

Mac OS-X systems

1. Install perl.
2. Install CPAN.
3. Set FTP into passive mode so that CPAN can use it.
4. In your bash shell, or your `~/.bash_login` file, add the following line:
`export FTP_PASSIVE=1`
5. Log into CPAN, and use it to install XML-RPC:
`cpan[2]> get XML::RPC`

```
cpan[2]> install XML::RPC
```

Creating the XML-RPC Object

As shown in the sample code below, the first step is to create an XML-RPC object that includes the WAN IP address of the Barracuda Load Balancer, password, and timeout value. The request is an HTTP POST to /cgi-mod/api.cgi:

```
http://<Barracuda Load Balancer WAN IP address: port>/cgi-  
mod/<api>?password=API password
```

For improved security, using https (default port 443) for sending the password is recommended:
https://<Barracuda Load Balancer WAN IP address>/cgi-mod/<api>?password=API
password

```
#!/usr/bin/perl use strict; use warnings; use XML::RPC; use LWP::UserAgent;  
use Data::Dumper; my $ua = LWP::UserAgent->new ; $ua->timeout( 60 ) ; #  
Create the XML::RPC object my $xmlrpc = XML::RPC->new  
( 'https://10.5.126.63/cgi-mod/api.cgi?password=admin', 'lwp_useragent' => $ua  
);
```

Return Codes

If there is an error, the XML-RPC new call returns a hash reference containing two entries: a fault code and a descriptive string.

If the password given is wrong, the hash returned is:

faultCode	faultStr
403	Access denied: Configuration administration password mismatch.

If the caller IP address is not within the allowed API/IP range, the hash returned is:

faultCode	faultStr
403	Access denied: Your machine does not have access to administer configuration.

For all of the API calls described in the rest of this document except for the service.show API, the hash returned is a value and a message:

```
$result = $xmlrpc->call( ... );  
$val = $$result{val};
```

```
$msg = $$result{msg};
```

The possible value and message pairs returned are listed below.

Successful Results

The output of a successful call is a hash with a value and a message. The message varies depending on the call made:

val	msg
200	Service successfully added OR Server successfully added OR Server successfully deleted OR Service successfully deleted OR Server mode successfully changed

Failure Results

The output of a failed call is a hash with a value and a message. The msg column in this table shows some of the returned messages:

val	msg
700	Invalid parameter in configuration
702	ERROR: Service already exists OR ERROR: Server already exists OR ERROR: (passed IP Address) does not exist OR ERROR: Service does not exist
704	ERROR: (passed value) is an invalid action
707	This operation is not permitted until this Barracuda Load Balancer has been activated OR Error in the parameters passed, e.g., ERROR: Illegal value (passed value) for Real Server: must be in IP address notation.

Related Articles

- [API Descriptions](#)
- [Sample Program - Multiple Functions](#)
- [Sample Program - Changing Real Server Status to Maintenance Mode](#)

© 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.