

Sample Program - Changing Real Server Status to Maintenance Mode

<https://campus.barracuda.com/doc/3539061/>

This program changes the status for a list of Real Servers to maintenance mode. It performs the following functions:

- For each IP address in a list, it sets the status of the Real Server to maintenance. The Barracuda Load Balancer will not direct any new client connections or requests to a Real Server in maintenance mode;
- It displays the Real Server state to confirm that the state has changed.

Edit the IP addresses – 10.5.126.63 should be the WAN IP address of your Barracuda Load Balancer, 192.168.132.214 is the VIP address of an existing Service, and 15.15.15.11 and 15.15.15.12 are the IP addresses of the Real Servers to be put into maintenance mode. You can put more Real Servers in the \$server_ips list, just be sure to separate them by a comma (,). Change the password to the API password for the Barracuda Load Balancer.

You can also modify this program to disable the Real Servers or make them active again by changing the value of the \$action parameter to disable or enable.

To run `maint.pl`, at a command prompt type:
`perl maint.pl`

`maint.pl`

```
#!/usr/bin/perl

use strict;
use warnings;

use XML::RPC;
use LWP::UserAgent;

# Service parameters
my $service_ip = '192.168.132.214' ;
my $service_protocol = 'TCP' ;
my $service_port = '80' ;
```

```
# Server parameters
my $server_ips = '15.15.15.11,15.15.15.12' ;
my $server_port = '80' ;

my $method_change = "server.change_state" ;
my $action = 'maintenance' ; #disable/enable/maintenance
my $service_id = "$service_ip:$service_port:$service_protocol" ;
my @server_ip = split(",", $server_ips);
my $service_show = 'status/state' ;
my $result;

# Create the XML::RPC object
my $ua = LWP::UserAgent->new ;
$ua->timeout( 60 );
my $xmlrpc = XML::RPC->new (
    'https://10.5.126.63/cgi-mod/api.cgi?password=admin',
    'lwp_useragent' => $ua );

#Change Real Server status for the IP addresses in the list
#to maintenance mode

my $disable_parameters = {
    vip => $service_id,
    port => $server_port,
    action => $action };

foreach my $ip (@server_ip){
    $disable_parameters->{'ip'} = $ip;
    print "Setting status for server $ip to $action.\n";
    $result = $xmlrpc->call($method_change, $disable_parameters );
    print "$result->{'msg'}\n";

    #Show the Real Server state now
    $result = $xmlrpc->call( 'service.show',
        { vip => $service_id,
          port => $server_port,
          ip => $ip,
          show => $service_show
        }
    );
    my $instance_id = "$ip" . ":" . "$server_port";
    print
    "Server status: $result->{$service_id}{$instance_id}{'state'}\n";
}
```

Related Articles

- [Automation API Guide](#)
- [API Descriptions](#)
- [Sample Program - Multiple Functions](#)

© 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.