Public Cloud Provisioning Command Line Tools

https://campus.barracuda.com/doc/96026703/

The following command-line tools are available to automate configuration tasks during deployment of your CloudGen Firewall:

- getpar
- cloud-restore-license
- create-dha
- editconf
- cloud-set-boxip
- cloud-enable-webui
- cloud-enable-ssh

Retrieving PAR files from a Firewall Control Center

Use this command to retrieve PAR files from a Control Center during provisioning. For PAYG firewalls, the licenses are pushed to the Control Center before fetching the configuration.

Usage: **getpar -a <CC IP address> -c <clustername> -r <range id number> -b <firewall name>**

- **-a|--address <address>** – Control Center IP address.
- **-u|--username <username>** – CC admin user used to connect to the Control Center.
- **-c|--cluster <cluster>** – Cluster name.
- **-r|--range <range>** – Range number.
- **-b|--boxname <boxname>** – Firewall name.
- **-d|--destination <dest>** – Destination directory and filename for the PAR file. E.g., /opt/phion/update/box.par
- **-s|-spoe** – Use Single Point of Entry to connect to the Control Center.
- **-l|--pushlic auto|always|never** – Configures if the licenses should be pushed to the Control Center before retrieving the PAR file. For PAYG firewalls, the license must be pushed to the Control Center.

For more information, see How to Modify CloudFormation Templates to Retrieve the PAR File from a Control Center.

Restore the Auto-Generated License on a PAYG Firewall

When you restore a PAYG-licensed firewall in the cloud using a PAR file from another PAYG instance, the license's files are overwritten with the licenses in the PAR file, resulting in an unlicensed firewall. By running this CLI command after restoring from a PAR file, the PAYG license generated during provisioning is restored, overwriting the existing license. This allows the admin to automatically restore a PAYG unit from a PAR file without having to back up and restore the license manually.

Usage: **cloud-restore-license**

Available parameters:

- **-f** – force license overwrite.
- **-h** – print this help and exit.

Create a High Availability Cluster

Execute this command on the primary firewall to create a high availability cluster via command line. You are prompted for the password for the other firewall. If the secondary firewall is running in the public cloud, you must disable enforcing a password change on the secondary firewall by adding the following editconf commands to the provisioning / user data scripts:

Usage: **/opt/phion/bin/create-dha -c -o <IP address of other firewall> -g <IP address for default gateway used by the other firewall>**

```
/opt/phion/bin/editconf -f /opt/phion/config/active/boxadm.conf -p
RPASSWDENFORCE -v 0
/opt/phion/bin/editconf -f /opt/phion/config/configroot/boxadm.conf -p
RPASSWDENFORCE -v 0
```

HA pairs in the cloud must be in subnets of the same network size. If you are using your own templates, the templates will no longer work in firmware version 8.0.1 and therefore must be updated.

Available parameters:

- **-u|--username <username>** – Specify username for connecting to the secondary firewall (default: root).
- **-o|--other-ip <address>** – IP address of the secondary firewall.
- **-g|--other-gw <address>** – IP address of the default gateway for the subnet in which the secondary firewall is running.
- **-s|--server <server name>** – (Optional) Specify the virtual server name used for the high availability cluster on devices where a virtual server still exists (e.g., on devices with upgraded firmware).
- **-c|-cleardirty** – Clear the dirty download flag after setting up the high availability cluster.

- **--verbosity <verbosity>** – Enable command-line logging, and set verbosity to the specified level.
- **--fullcolortrace** – Enable colored command-line logging.

For more information on high availability, see High Availability.

Insert or Edit a Configuration Parameter

In some cases, you may be required to edit a configuration parameter. For example, you need to disable enforcing a password change on first login when pairing a high availability cluster via **create-dha**.

Usage: **editconf -f </absolutepath/file.conf> -p <parameter to set> -v <value for the parameter>**

- **-f|--file <input config file, absolute path>** – Absolute path to the configuration file.
- **-p|--put <key to set in the config file>** – Set a configuration parameter. Also requires value to be set.
- **-d|--delete <key to delete>** – Delete a configuration parameter.
- **-D|--delete-section** – Remove the entire section from the configuration file.
- **-v|-value <value content>** – Value to add to the configuration file.

Configure the Network on Initial Boot

This tool allows you to configure IP addresses on different interfaces and to introduce routes on initial boot, e.g., setting the management IP, adding an additional IP, and adding a route.

> If not otherwise specified, the tool **cloud-set-boxip** sets the stated IP address as management IP for eth0 with a subnet of 24, an MTU of 1500, and the gateway x.x.x.1.
>
> If you are specifying the management IP address, DHCP will be deactivated automatically.

Usage: **cloud-set-boxip <argument>**

- **-h| --help** – Shows the help.
- **-d <DEVICE> | --device <DEVICE>** – Enter to specify the interface, e.g., `-d eth1`.
- **-a <ADDRESS> | --address <ADDRESS>** – Sets the IP address.
- **-n <NETMASK>| --netmask <NETMASK>** – Sets the netmask.

- **-g <GATEWAY>| --gateway <GATEWAY>** – Sets the gateway.
- **--dry-run** – This allows you to try a command. If you use **--dry-run**, the command before will not be executed and prompts what the CloudGen Firewall would do if the command is used without **--dry-run,** e.g.:
  `cloud-set-boxip -a 10.10.8.1 -d eth1 --dry-run` will tell you that it would set the IP address for eth1 to 10.10.8.1.
- **--mip <true/false/auto>** – Specify whether the IP address should be a management IP address. Valid arguments are true, false, and auto.
- **--activate** – Activates the configuration (same function as **Activate** in Barracuda Firewall Admin).
- **--mtu MTU** – Set a value for the MTU.
- **--route** – Creates a new route. Needs additional parameters such as route name, interface, and network, e.g., `cloud-set-boxip --route --route-name droute001 -d eth0 -a 192.168.0.0/24.`
- **--route-name <ROUTE_NAME>** – Specify the name of the created route.
- **--foreign** – If enabled, any IP address active on the device will suffice to bring up the route.
- **--name <NAME_OF_ADDITIONAL_IP>** – Specify the name of the additional IP.

For multiple invocations, the MTU must always be the same! Using devices with different MTUs is not supported.

Example Usage:

```
# set MIP on eth0 to 10.0.0.10/24
$ cloud-set-boxip -a 10.0.0.10
# add first additional IP as 192.168.0.1/25
$ cloud-set-boxip -a 192.168.0.1 -n 25 -d eth1
# add second additional IP as 172.16.1.123/24 and activate
$ cloud-set-boxip -a 172.16.1.123 -d eth2 --activate
# add a new device / direct route
$ cloud-set-boxip --route --route-name droute001 -d eth0 -a 192.168.0.0/24
# add a new gateway route
$ cloud-set-boxip --route --route-name gw001 -g 1.2.3.4 -a 192.168.1.0/24
```

Enable the Web Interface on Public Cloud Firewalls

The web interface is disabled automatically if user data scripts are used in the template. To manually re-enable the web interface, use **/opt/phion/bin/cloud-enable-webui**

Usage: **/opt/phion/bin/cloud-enable-webui**

Enable SSH Access on Public Cloud Firewalls

By default, SSH access for the root user is disabled for firewalls running in the public cloud. You can re-enable SSH via **cloud-enable-ssh**

Usage: **cloud-enable-ssh**

Microsoft Azure Virtual WAN Connection

Use the commands presented here to establish a connection to Microsoft Azure Virtual WAN manually via a commend-line interface script. The script will also start the connection daemon.

Usage: **/opt/phion/bin/connectToAzureVWAN <tenant_id> <client_id> <client_secret> <subscription_id> <ng_user> <ng_password> <vwan_name>**

If the daemon was killed for whatever reason, it must be restarted. The input values are handed over to the daemon directly by *n<value>* arguments.

Usage: **echo "<tenant_id>\n<client_id>\n<client_secret>\n<subscription_id>\n<ng_user>\n<ng_password>\n<resource_group>\n<vwan_name>\n<site_id>" | /opt/phion/bin/azureVWANDaemon**