phionctrl

https://campus.barracuda.com/doc/41116382/

To manage routing, IP addresses, interfaces, firewall processes, services, and modules for the Barracuda NG Firewall, use the ***phionctrl*** utility.

The following ***phionctrl*** commands are available:

Command	Usage
phionctrl route	To display all active IP addresses, gateways, main routes, VPN interfaces, and the IP addresses of the running VPN service.
phionctrl server	To manage and monitor running servers.
phionctrl service	To manage services on a specific server without shutting down all available services.
phionctrl module	To manage software modules.
phionctrl ip	To manage IP addresses.
phionctrl arp	To detect duplicate IP addresses in the network.
phionctrl tell	To send unsolicited ARP requests.
phionctrl proc show	To view information about processes.
phionctrl hostid	To display the IDs of hardware components.
phionctrl lic	To display license information for modules.
phionctrl session	To view and kill management sessions.
phionctrl usage	To monitor the CPU usage of all processes during a specified interval of time in milliseconds.
phionctrl box	To monitor and manage processes that are specific to the Barracuda NG Firewall (and not the operating system).
phionctrl versions	To display the versions for modules.
phionctrl startup	To start the Barracuda NG Firewall subsystem (operating system) and its servers and services
phionctrl shutdown	To shut down the Barracuda NG Firewall subsystem (operating system) and its servers and services

phionctrl route

To display all active IP addresses, gateways, main routes, VPN interfaces, and the IP addresses of the running VPN service, use the following command:

phionctrl route show

Example Usage

The following table displays example output for the *phionctrl route show* command:

```
root@HQ-NG1:~]# phionctrl route show ---------- Active IPs ---------------
10.0.10.61/0 eth0 UP 00-0c-29-22-84-70 10.0.10.88/7 eth0 UP 00-0c-29-22-84-70
127.0.0.1/8 lo UP 00-00-00-00-00-00 127.0.3.1/8 pvpn0 vpn0 vpnr0 UP
00-00-00-00-00-00 172.16.0.254/0 eth3 UP 00-0c-29-22-84-8e 194.93.0.195/8
dhcp UP 00-0c-29-22-84-84 62.99.0.40/0 eth1 UP 00-0c-29-22-84-7a ----------
Active Routing Tables ---- vpnlocal 0 up device 192.168.0.0/8 dev pvpn0 src
0.0.0.0 metric 0 table vpnlocal foreign Name= 5 0 POLICY from 10.0.11.0/8 up
device 172.16.0.0/8 dev vpn0 src 0.0.0.0 metric 0 table 5 foreign Name= dhcp1
0 POLICY from 194.93.0.195/0 up device 194.93.0.0/8 dev dhcp src 194.93.0.195
metric 0 table dhcp1 foreign Name= main 0 up device 194.93.0.0/8 dev dhcp src
194.93.0.195 metric 0 table main foreign Name= up device 62.99.0.0/8 dev eth1
src 62.99.0.40 metric 0 table main Name=HQ-ISP1 up device 172.16.0.0/8 dev
eth3 src 172.16.0.254 metric 0 table main Name=HQ-DMZ up device 127.0.3.0/8
dev pvpn0 src 127.0.3.1 metric 0 table main foreign Name= up device
127.0.3.0/8 dev vpnr0 src 127.0.3.1 metric 0 table main foreign Name= up
device 127.0.3.0/8 dev vpn0 src 127.0.3.1 metric 0 table main foreign Name=
up device 10.0.10.0/7 dev eth0 src 10.0.10.88 metric 0 table main foreign
Name=boxnet up device 194.93.0.254/0 dev dhcp src 194.93.0.195 metric 0 table
main foreign Name= HQ-ISP1 0 POLICY from 62.99.0.0/8 up gateway 0.0.0.0/32
dev eth1 via 62.99.0.254 src 62.99.0.40 metric 0 table HQ-ISP1 foreign
Name=HQ-ISP1a default 0 up gateway 0.0.0.0/32 dev eth1 via 62.99.0.254 src
62.99.0.40 metric 1 table default Name=HQ-ISP1a ---------- Active v6 IPs ----
---------- ---------- Active v6 Routing Tables ---- main 32767 POLICY from
all/0 to all/0 [2014-03-20 16:11 CET] [-root shell-] [-Barracuda Networks-]
[root@HQ-NG1:~]#
```

Typically, i nformation is dumped to the display with standard output (stdout). If necessary, you can also pipe information to a file. To pipe information to a file, append the following to the command:

> /path/filename

For example, if you want to write the output of a command to a file in the */tmp* directory:

[root@mybox:~] phionctrl route show > /tmp/route

The piping function might facilitate error localization. If you experience any problems, pipe the command output to a file and email the file to Barracuda Networks Technical Support.

phionctrl server

To manage and monitor running servers, use the *phionctrl server* command. Use the following syntax:

phionctrl server <option> [server-name]

You can use the following options with this command:

Option	Description
show	Displays the state and configuration of the server. The ***show*** option is useful for verifying that servers have been started, stopped, blocked, and restarted. Possible server states include the following: • **down**—The server is not running at the moment. • **primary/secondary**—The server is running as a primary or secondary box in a high availability (HA) environment. • **blocked**—The server is blocked. The *active* parameter in the command output specifies if the server is active or inactive. Possible values for this parameter include: • **0**—The server is inactive. • **1**—The server is active.
start	Starts the specified server. For example, to start a server named mc: *phionctrl server start mc*
stop	Stops the specified server and all of its services. For example, to stop a server named mc: *phionctrl server stop mc* The control daemon will restart the stopped server within a few seconds. To stop the server permanently, use the ***block*** option instead.
restart	When necessary, use this option to restart the server and its services (e.g. after making configuration changes). For example, to restart a server named mc: *phionctrl server restart mc* You may verify the control daemon managed restarting function by sending the ***stop*** option to the server and then reviewing the server and services getting restarted automatically.
block	Blocks the specified server so that the control daemon will not restart it. The server and all of its services are permanently stopped. For example, to block a server named mc: *phionctrl server block mc*
unblock	Unblocks the specified server. For example, to unblock a server named mc: *phionctrl server unblock mc* An unblocked server does not automatically start. It remains down until you send the ***start*** option to it. To unblock a server and have the control daemon restart it automatically within a few seconds, use the ***stop*** option instead of the ***unblock*** option.

Example Usage

This section provides examples of how to use the *phionctrl server* command to manage a server named mc.

1. Display the state of the server. The following example output shows that the server is blocked and inactive.
   ```
   [root@ash:~]# phionctrl server show mc state=block active=0
   other=unknown task=primary Box: ash(10.0.10.10) Server IPs: 10.0.10.11
   Active IPs: Server Services: Conf DNS Event Log PKI StatC StatV VPN mFW
   Active Services: Blocked Services:
   ```
2. Unblock the server.
   ```
   [root@ash:~]# phionctrl server unblock mc
   ```
3. Verify that the server is unblocked. The following example output shows that the server is no longer blocked, but it is still down.
   ```
   [root@ash:~]# phionctrl server show mc state=down active=0 other=unknown
   task=primary Box: ash(10.0.10.10) Server IPs: 10.0.10.11 Active IPs:
   Server Services: Conf DNS Event Log PKI StatC StatV VPN mFW Active
   Services: Blocked Services:
   ```
4. Start the server.
   ```
   [root@ash:~]# phionctrl server start mc
   ```
5. Verify that the server has been started. The following example output shows that the server is running.
   ```
   [root@ash:~]# phionctrl server show mc state=primary active=1
   other=unknown task=primary Box: ash(10.0.10.10) Server IPs: 10.0.10.11
   Active IPs: 10.0.10.11 Server Services: Conf DNS Event Log PKI StatC
   StatV VPN mFW Active Services: Conf DNS Event Log PKI StatC StatV VPN
   mFW Blocked Services:
   ```

phionctrl service

To manage services on a specific server without shutting down all available services, use the *phionctrl service* command. Use the following syntax:

phionctrl service <option> [server-name] [service-name]

You can use the following options with this command:

Option	Description
show	Displays all servers and their running services on a Barracuda NG Firewall. The ***show*** option is useful for verifying that services have been started, stopped, blocked, and restarted.

start	Starts a service manually. If the service is not blocked, it is started automatically by the control daemon. For example, to start the DNS service on a server named mc: *phionctrl service start mc DNS*
stop	Stops a service on a specific server. If the service has not been blocked, it is later started automatically by the control daemon. For example, to stop the DNS service on a server named mc: *phionctrl service stop mc DNS*
restart	Restarts a service on a specific server. You may need to restart a service after making manual configuration file changes. For example, to restart the DNS service on a server named mc: *phionctrl service restart mc DNS*
block	Blocks a service so that is not started automatically by the control daemon. For example, to block the DNS service on a server named mc: *phionctrl service block mc DNS* To start the service later, you can either use the ***start*** option or unblock the service with the ***stop*** option. The control daemon then starts the service automatically.

Example Usage

This section provides examples of how to use the *phionctrl service* command to manage services on a server named mc.

1. Display the state of the services. The following example output shows that all services are up.
   ```
   [root@ash:~]# phionctrl service show server mc Conf up numProc=7
   numFD=113 mem=15140kB DNS up numProc=2 numFD=14 mem=2080kB
   ```
2. Block the DNS service.
   ```
   [root@ash:~]# phionctrl service block mc DNS
   ```
3. Verify that the DNS service has been blocked. The following example output shows that the DNS service is blocked.
   ```
   [root@ash:~]# phionctrl service show server mc Conf up numProc=7
   numFD=113 mem=15140kB DNS block numProc=0 numFD=0 mem=0kB
   ```

phionctrl module

With the ***phionctrl module*** command, you can manage the following software modules:

- **firewall**
- **cfirewall**
- **dhcpe**
- **dhcprelay**
- **ftpgw**

- **ospf**
- **policyserver**
- **spamfilter**
- **sshprx**
- **virscan**
- **vpnserver**
- **dns**
- **snmp**
- **proxy**
- **cfirewall**
- **mailgw**

Use the following syntax:

phionctrl module <option> [module-name]

You can use the following options with this command:

Option	Description
show	Displays the state of the specified software module. For example, to view the state of the firewall module: *phionctrl module show firewall*
start	Starts all services bound to the specified module. For example, to start the firewall module: *phionctrl module start firewall*
stop	Stops the specified module. If the module was not blocked, it is then restarted by the control daemon. For example, to stop the dns module: *phionctrl module stop dns*
restart	Restarts the specified software module. For example, to restart the dns module: *phionctrl module restart dns*
block	Blocks the specified software module. If the software module is blocked, the corresponding services are not restarted by the control daemon. For example, to block the dns module: *phionctrl module block dns* To later restart the blocked module, use the ***start*** option.

Example Usage

This section provides examples of how to use the *phionctrl module* command to manage the DNS module.

1. Display the state of the DNS module. The following example output shows that the module is up.
   ```
   [root@ash:~]# phionctrl module show dns server mc DNS up numProc=0
   numFD=0 mem=0kB
   ```

2. Block the DNS module.
   ```
   [root@ash:~]# phionctrl module block dns
   ```
3. Verify that the DNS module has been blocked. The following example output shows that the module has been blocked.
   ```
   [root@ash:~]# phionctrl module show dns server mc DNS block numProc=0 numFD=0 mem=0kB
   ```
4. Start the DNS module.
   ```
   [root@ash:~]# phionctrl module start dns
   ```
5. Verify that the DNS module has been started. The following example output verifies that the module is up.
   ```
   [root@ash:~]# phionctrl module show dns server mc DNS up numProc=0 numFD=0 mem=0kB
   ```

phionctrl ip

To manage IP addresses, use the *phionctrl ip* command. Use the following syntax:

phionctrl ip <option> [ip-address]

You can use the following options with this command:

Option	Description
show	Displays all active IP addresses and active routing tables.
add	Adds the specified IP address. For example, to add the 10.0.10.12 IP address: *phionctrl ip add 10.0 . 10.12* The corresponding interface is configured via the network. Otherwise, if no corresponding network can be found, the IP address is added to the loopback interface.
del	Deletes the specified IP address from the system. For example, to delete the the 10.0.10.12 IP address: *phionctrl ip del 10.0.10.12*

Example Usage

This section provides examples of how to use the *phionctrl ip* command to manage IP addresses.

1. Display all active IP addresses and active routing tables.
   ```
   [root@ash:~]# phionctrl ip show ---------- Active IPs ----------
   10.0.10.10/8 eth0:mip0 tap1 UP 00-0e-0c-4e-48-62 10.0.10.11/0 eth0:mc UP 00-0e-0c-4e-48-62 127.0.0.1/8 lo:loop UP 00-00-00-00-00-00 127.0.1.1/8 tap0:fw UP fe-fd-00-00-00-00 127.0.2.1/8 tap1 UP fe-fd-00-00-00-00
   ```

```
127.0.3.1/8 tap2:vpnpers UP fe-fd-00-00-00-00 169.254.1.11/0 tap2:aux2
UP fe-fd-00-00-00-00 ---------- Active Routing Tables ---- vpnlocal 0 up
device 10.0.10.208/4 dev tap1 src 0.0.0.0 metric 0 table vpnlocal
foreign Name= main 0 up gateway 172.16.16.0/8 dev eth0 via 10.0.10.196
src 10.0.10.10 metric 0 table main Name=arztest up device 127.0.1.0/8
dev tap0 src 127.0.1.1 metric 0 table main foreign Name= up device
127.0.3.0/8 dev tap2 src 127.0.3.1 metric 0 table main foreign Name= up
device 127.0.2.0/8 dev tap1 src 127.0.2.1 metric 0 table main foreign
Name= up gateway 172.16.10.0/8 dev eth0 via 10.0.10.22 src 10.0.10.10
metric 0 table main Name=172-1 up device 10.0.10.0/8 dev eth0 src
10.0.10.10 metric 0 table main foreign Name=boxnet default 0 up gateway
0.0.0.0/32 dev eth0 via 10.0.10.1 src 10.0.10.10 metric 0 table default
Name=boxdev
```

2. Add the 10.0.10.12 and 10.0.2.200 IP addresses.
   ```
   [root@ash:~]# phionctrl ip add 10.0.10.12 [root@ash:~]# phionctrl ip add
   10.0.2.200
   ```
3. Verify that the 10.0.10.12 and 10.0.2.200 IP addresses have been added. As displayed in the following example output, 10.0.10.12 binds to the eth0 interface because the 10.0.10.0/8 network belongs to this interface. The 10.0.2.200 IP address binds to the loopback interface because no corresponding network can be found.
   ```
   [root@ash:~]# phionctrl ip show ---------- Active IPs --------------
   10.0.10.10/8 eth0:mip0 tap1 UP 00-0e-0c-4e-48-62 10.0.10.11/0 eth0:mc UP
   00-0e-0c-4e-48-62 10.0.10.12/0 eth0: UP 00-0e-0c-4e-48-62 10.0.2.200/0
   lo: UP 00-00-00-00-00-00 127.0.0.1/8 lo:loop UP 00-00-00-00-00-00
   127.0.1.1/8 tap0:fw UP fe-fd-00-00-00-00 127.0.2.1/8 tap1 UP fe-
   fd-00-00-00-00 127.0.3.1/8 tap2:vpnpers UP fe-fd-00-00-00-00
   169.254.1.11/0 tap2:aux2 UP fe-fd-00-00-00-00
   ```
4. Delete the 10.0.10.12 and 10.0.2.200 IP addresses.
   ```
   [root@ash:~]# phionctrl ip del 10.0.10.12 [root@ash:~]# phionctrl ip del
   10.0.2.200
   ```
5. Verify that the the 10.0.10.12 and 10.0.2.200 IP addresses have been deleted. The following example output shows that the IP addresses have been deleted and are no longer listed.
   ```
   [root@ash:~]# phionctrl ip show ---------- Active IPs --------------
   10.0.10.10/8 eth0:mip0 tap1 UP 00-0e-0c-4e-48-62 10.0.10.11/0 eth0:mc UP
   00-0e-0c-4e-48-62 127.0.0.1/8 lo:loop UP 00-00-00-00-00-00 127.0.1.1/8
   tap0:fw UP fe-fd-00-00-00-00 127.0.2.1/8 tap1 UP fe-fd-00-00-00-00
   127.0.3.1/8 tap2:vpnpers UP fe-fd-00-00-00-00 169.254.1.11/0 tap2:aux2
   UP fe-fd-00-00-00-00
   ```

phionctrl arp

To detect duplicate IP addresses on your network, use the *phionctrl arp* command. You can detect duplicate IP addresses either for a specific IP address or for all configured IP addresses in the network.

Use the following syntax:

phionctrl arp <ip-address> | all

The command uses the ARP protocol to assign an IP address to the physical address of a network card (MAC address). If a duplicate IP address is found, an error message related to the corresponding MAC address is displayed.

Example Usage

This section provides examples of how to use the *phionctrl arp* command.

1. Search for any duplicates for the 10.0.10.10 IP address. The following example output shows that no duplicate IP addresses have been detected.
   `[root@ash:~]# phionctrl arp 10.0.10.10 no duplicate IPs detected`
2. Search for any duplicates for all configured IP addresses. The following example output shows that no duplicate IP addresses have been detected.
   `[root@ash:~]# phionctrl arp all probe 10.0.10.10\probe 10.0.10.11 ----------no duplicate IPs detected`

phionctrl tell

The ARP protocol is a passive protocol. For example, a network interface will remain silent until an ARP request is received. To send unsolicited ARP requests, use the *phionctrl tell* command. Use the following syntax:

phionctrl tell <ip-address>

Example Usage

The following table displays an example of how to send unsolicited ARP requests to the 10.0.10.10 IP address.

```
[root@ash:~]# phionctrl tell 10.0.10.10 send unsolicited ARP for 10.0.10.10 to 10.0.10.255 on eth0
```

phionctrl proc show

To view information about processes, use the *phionctrl proc show* command. You view information for all processes, a specific process name, or a process ID. Use the following syntax:

phionctrl proc show all | [process-name] | [pid]

Example Usage

This section displays examples of how to use the *phionctrl proc show* command.

1. View information for the controld process.
   ```
   [root@ash:~]# phionctrl proc show controld 6 processes: 2640 2664 2675
   10225 751 3306 35 file descriptors 2312 kB Memory 2120 kb shared Memory
   Open Files: /dev/null /proc/2907/statm Listening Sockets: 10.0.10.10:801
   Established Sockets: 10.0.10.10:801->10.0.4.136:1729 UDP Sockets:
   0.0.0.0:32946 10.0.10.10:32944 10.0.10.10:801 127.0.0.1:32965
   127.0.0.1:32971
   ```
2. View information for PID 2495.
   ```
   [root@ash:~]# phionctrl proc show 2495 1 processes: 2495 13 file
   descriptors 276 kB Memory 1224 kb shared Memory Open Files: /dev/acpf
   /dev/null
   ```

phionctrl hostid

To display the IDs of hardware components, such as the CPU ID, MAC addresses, and motherboard ID, use the *phionctrl hostid* command. This information is necessary for licensing purposes.

Example Usage

The following table displays example output for the *phionctrl hostid* command.

```
[root@ash:~]# phionctrl hostid CPU-0000-0F29-003B-7040-0000-0000 BBS-
BZTP44000670 MAC-00:0e:0c:4e:48:62 MAC-00:0e:0c:4e:48:63
```

phionctrl lic

To display license information, use the *phionctrl lic* command. You can display information either for

all licenses or for a specific module. Use the following syntax:

phionctrl lic [module-name]

If a module name is entered, the specific license is displayed. A license is often issued for multiple services. If this is the case, then the scope of modules covered by the license is displayed in the subsection.

Example Usage

The following table displays example output for viewing information for all licenses.

```
[root@ash:~]# phionctrl lic ----------------------------------------license
= 000000AT001-MC-ES-131 hostid = MAC-00:0e:0c:4e:48:62 module = base-mces
Private key is set grace = 2 policy = 0 version = 1 password is NOT present
Issuer_C = AT Issuer_CN = Sales Issuer_L = Innsbruck Issuer_O = Barracuda
Networks Issuer_OU = Barracuda Networks Inc. Issuer_ST = Tirol Subject_C = AT
Subject_CN = Barracuda Networks Inc. Subject_L = Innsbruck Subject_O = Cuda
Subject_unstructuredName = grace:2 id:MAC-00:0e:0c:4e:48:62 lic:000000AT001-
MC-ES-131 mod:base-MCES protip:0 sub:firewall,
dns,rangeconf,dstatm,qstatm,mevent,mastervpn,pki grace = 2 id =
MAC-00:0e:0c:4e:48:62 lic = 000000AT001-MC-ES-131 mod = base-MCES protip = 0
sub = firewall,dns,rangeconf,dstatm,qstatm,mevent,mastervpn,pki Costumer:
Country = AT State = Organisation = Cuda Org. Unit = Name = Cuda Email =
Issuer: Country = AT State = Tirol Organisation = Cuda Org. Unit = Cuda Name
= Sales
```

phionctrl session

To view and kill management sessions on a Barracuda NG Firewall, use the *phionctrl session* command. Use the following syntax:

phionctrl session <option>

You can use the following options with this command:

Option	Description
show	Displays all open sessions on a Barracuda NG Firewall and their PIDs.
kill <pid>	Kills a management session for the specified PID.

phionctrl usage

To monitor the CPU usage of all processes during a specified interval of time in milliseconds, use the *phionctrl usage* command. Use the following syntax:

phionctrl usage <interval-in-milliseconds> [r]

To also display all process names and split them into single PIDs, add the ***r*** option. The ***r*** option is useful for detecting a process that may be blocking the system.

Example Usage

This section provides examples of how to use the use the *phionctrl usage* command.

1. View CPU usage for all processes during an interval of 10 milliseconds.
```
[root@ash:~]# phionctrl usage 10 bash 0 0 0 bdflush 0 0 0 bdns 0 0 0
boxconfigd 0 0 0 bsyslogd 0 0 0 bsyslogd_slgd 0 0 0 controld 100 30 70
crond 0 0 0 cstatd 30 10 20 distd 0 0 0 eventd 0 0 0 fwauthd 0 0 0 gpm 0
0 0 init 0 0 0 keventd 0 0 0 khubd 0 0 0 kjournald 10 0 10
ksoftirqd_CPU0 0 0 0 kswapd 0 0 0 kupdated 0 0 0 logd 0 0 0 logwrapd 0 0
0 masterd 0 0 0 mc_Conf 30 30 0 mc_DNS 0 0 0 mc_Event 0 0 0
```
2. View CPU usage for all processes during an interval of 10 milliseconds and add the ***r*** option to also display all process names and split them into single PIDs.
```
[root@ash:~]# phionctrl usage 10 r arztest.sh@25562 0 0 0 bash@25874 0 0
0 bdflush@5 0 0 0 bdns@18855 0 0 0 boxconfigd@2749 0 0 0 boxconfigd@4062
0 0 0 bsyslogd@2833 0 0 0 bsyslogd_slgd@2987 0 0 0 controld@10225 90 70
20 controld@2640 0 0 0 controld@2664 0 0 0 controld@2675 0 0 0
controld@751 0 0 0 controld@8261 10 10 0 crond@25559 0 0 0 crond@402 0 0
0 cstatd@2828 0 0 0 cstatd@2986 40 10 30 distd@2876 0 0 0 eventd@2935 0
0 0 eventd@3025 0 0 0 eventd@3026 0 0 0 eventd@3027 0 0 0 fwauthd@2495 0
0 0 gpm@2667 0 0 0 init@1 0 0 0 keventd@2 0 0 0 khubd@7 0 0 0
kjournald@12 10 0 10 kjournald@84 0 0 0 kjournald@85 10 0 10
ksoftirqd_CPU0@3 0 0 0 kswapd@4 0 0 0 kupdated@6 0 0 0 logd@2958 0 0 0
logwrapd@2982 0 0 0 mc_Conf@19876 0 0 0 mc_Conf@19884 0 0 0
```

phionctrl box

To monitor and manage processes that are specific to the Barracuda NG Firewall (and not the

operating system), use the *phionctrl box* command. Use the following syntax:

phionctrl box <option>

You can use the following options with this command:

Option	Description
show	Displays all processes that are specific to the Barracuda NG Firewall. This option is also useful for verifying that all daemons are up and running.
start <process>	Starts the specified process if it is down. If the process daemon is down and unblocked, it is also started by the control daemon.
stop <process>	Stops the specified process. If a service is blocked, it may be unblocked with this option. The control daemon then starts it again after a few seconds.
restart <process>	Restarts the specified process.
block <process>	Blocks the specified process. The process is not restarted by the control daemon until it is unblocked.

Example Usage

This section provides examples of how to use the *phionctrl box* command.

1. Display all processes. The following example output shows that the cstat process is blocked.
   ```
   [root@ash:~]# phionctrl box show bdns bdns up listen=0 numProc=1 numFD=4
   mem=1044kB boxconfig boxconfigd up listen=0 numProc=2 numFD=9 mem=1728kB
   boxfw trans7 up listen=0 numProc=13 numFD=87 mem=48796kB bsyslog
   bsyslogd up listen=0 numProc=1 numFD=4 mem=1016kB control controld up
   listen=0 numProc=6 numFD=34 mem=4424kB cstat cstatd block listen=0
   numProc=0 numFD=0 mem=0kB dist distd up listen=0 numProc=1 numFD=5
   mem=916kB
   ```
2. Start the cstat process.
   ```
   [root@ash:~]# phionctrl box start cstat
   ```
3. Verify that the cstat process is started. The following example output shows that the process has been started successfully.
   ```
   [root@ash:~]# phionctrl box show bdns bdns up listen=0 numProc=1 numFD=4
   mem=1044kB boxconfig boxconfigd up listen=0 numProc=2 numFD=9 mem=1728kB
   boxfw trans7 up listen=0 numProc=13 numFD=87 mem=48796kB bsyslog
   bsyslogd up listen=0 numProc=1 numFD=4 mem=1016kB control controld up
   listen=0 numProc=6 numFD=34 mem=4424kB cstat cstatd up listen=0
   numProc=2 numFD=9 mem=1872kB dist distd up listen=0 numProc=1 numFD=5
   mem=916kB
   ```

phionctrl versions

To display the versions for modules, use the *phionctrl versions* command. You can view the versions either for all modules or for a specific module. Use the following syntax:

phionctrl versions [module-name]

If a module name is entered, only the version for that module is displayed.

Example Usage

The following example output lists the versions for all modules on the system.

```
[root@ash:~]# phionctrl versions kernel 2.4.28-2.4.2.8 bdns R-2.4_V-2.4.2.5
Nov 3 2004 12:32:00 boxconfig R-2.4_V-2.4.2.22 May 18 2005 18:12:49 boxfw
R-2.4_V-2.4.2.109 Apr 29 2005 10:50:28 bsyslog R-2.4_V-2.4.2.7 Jun 28 2005
11:15:00 control R-2.4_V-2.4.2.14 Aug 4 2005 09:39:23 cstat R-2.4_V-2.4.1.7
Aug 24 2005 19:27:54 dist R-2.4_V-2.4.1.9 Oct 27 2004 13:53:56 event
R-2.4_V-2.4.1.37 May 12 2005 15:05:18 log R-2.4_V-2.4.1.7 Apr 14 2005
16:58:41 logwrap R-2.4_V-2.4.1.5 Nov 5 2004 11:33:57 phibs R-2.4_V-2.4.1.15
Apr 11 2005 09:45:36 psyslog R-2.4_V-2.4.1.4 Oct 20 2004 11:11:37 qstat
R-2.4_V-2.4.1.6 Apr 14 2005 16:51:54 dstats R-2.4_V-2.4.1.6 Nov 4 2004
09:20:03 logstor 2.2.4-6 Aug 05 2003 08:11:13 cfirewall R-2.4_V-2.4.1.1 Mar 4
2005 12:12:17 clusterconf R-2.4_V-2.4.2.22 May 18 2005 18:12:49 mevent
R-2.4_V-2.4.1.37 May 12 2005 15:05:18 proxy R-2.4_V-2.4.1.6 May 1 2005
18:41:04 qstatm R-2.4_V-2.4.1.6 Apr 14 2005 16:51:54 rangeconf
R-2.4_V-2.4.2.22 May 18 2005 18:12:49 snmp R-2.4_V-2.4.2.2 Jun 6 2005
12:48:49 spamfilter 2.4.2-4 Jun 01 2005 12:06:30 sshprx R-2.4_V-2.4.2.2 Apr
11 2005 15:15:00 vpnserver R-2.4_V-2.4.2.131 Aug 22 2005 21:03:48
```

phionctrl startup and shutdown

To start and shut down the Barracuda NG Firewall subsystem (operating system) and its servers and services, use the following commands:

Command	Descriptions
phionctrl startup	Starts the Barracuda NG Firewall, reads all configuration files from the */opt/phion/config/active* directory, and starts the daemons and services.

phionctrl shutdown	Shuts down all services and the operating system.