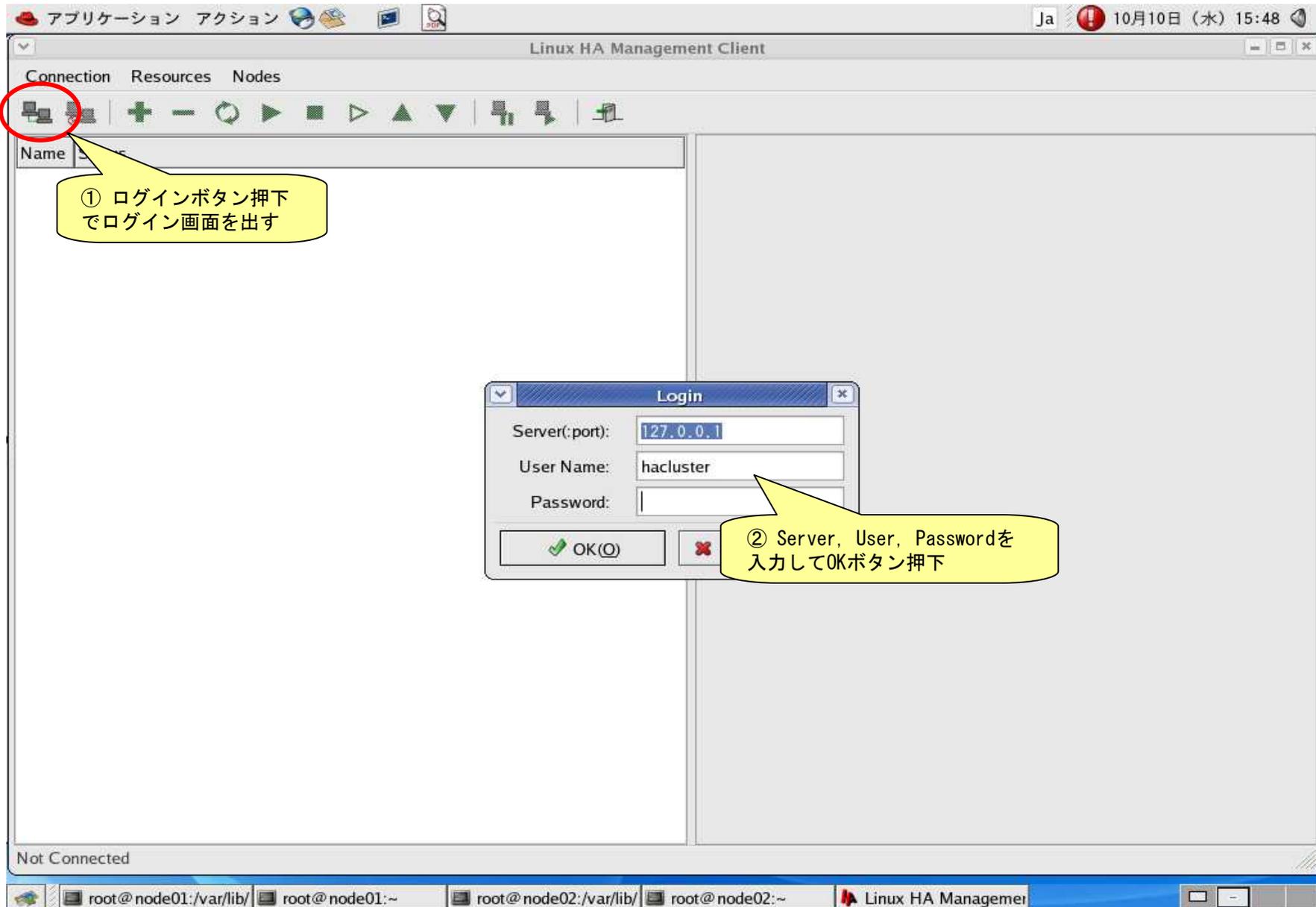


GUI ログイン



ログイン後の画面

The screenshot displays the Linux HA Management Client interface. The window title is "Linux HA Management Client". The top menu bar includes "Connection", "Resources", and "Nodes". Below the menu is a toolbar with various icons for cluster management. The main area is divided into two panes. The left pane shows a tree view of the cluster configuration:

Name	Status
linux-ha	with quorum
Nodes	
node02	running(dc)
node01	running
Resources	
Constraints	
Places	
Orders	
Colocations	

The right pane displays configuration parameters for the cluster:

- Version: 2.1.2
- Debug Level: 0
- UDP Port: 694
- Keep Alive: 2
- Warning Alive: 20
- Dead Time: 30
- Initial Dead Time: 60
- Symmetric Cluster
- Stonith Enabled
- Transition Timeout: 20s
- Resource Stickiness: 0
- No Quorum Policy: stop
- Resource Failure Stickiness: 0

At the bottom of the right pane are "Apply" and "Reset" buttons. The status bar at the bottom of the window indicates "Connected to 127.0.0.1". The taskbar at the very bottom shows several terminal windows and the Linux HA Manager icon.

クラスタの設定

The screenshot displays the Linux HA Management Client interface. On the left, a tree view shows the cluster configuration for 'linux-ha', including nodes (node01, node02), resources, and constraints. The right pane shows various configuration parameters for the cluster, such as Version (2.1.2), Debug Level (0), UDP Port (694), Keep Alive (2), Warning Alive (20), Dead Time (30), Initial Dead Time (60), Symmetric Cluster (checked), Stonith Enabled (unchecked), Transition Timeout (20s), Resource Stickiness (1000000), No Quorum Policy (ignore), and Resource Failure Stickiness (-1000000). A red oval highlights the 'No Quorum Policy' dropdown menu and the 'Apply' button. A yellow callout box with a speech bubble points to the 'Apply' button, containing the text: ① クラスタの設定を行い Applyボタン押下. The status bar at the bottom shows 'Connected to 127.0.0.1' and the system tray includes icons for root@node01, root@node02, and the Linux HA Manager.

Parameter	Value
Version	2.1.2
Debug Level	0
UDP Port	694
Keep Alive	2
Warning Alive	20
Dead Time	30
Initial Dead Time	60
Symmetric Cluster	<input checked="" type="checkbox"/>
Stonith Enabled	<input type="checkbox"/>
Transition Timeout	20s
Resource Stickiness	1000000
No Quorum Policy	ignore
Resource Failure Stickiness	-1000000

リソースグループの追加

Linux HA Management Client

Connection Resources Nodes

① 追加ボタンを押下し、ポップアップ画面で"Group"を選択。

② Group I Dを設定し、OKボタンを押下

Add Resource Group

ID: group01

Ordered: true

Collocated: true

OK(O) キャンセル(C)

No Data Available

Connected to 127.0.0.1

root@node01:/var/lib/ root@node01:~ root@node02:/var/lib/ root@node02:~ Linux HA Manager

仮想アドレスの追加

Linux HA Management Client

Connection Resources Nodes

Name Status

- linux-ha
 - Nodes
 - node02
 - node01
 - Resources
 - Constraints
 - Places
 - Orders
 - Colocations

① リソースIDを指定

Resource ID: ip01 Belong to group: group01 (type for new one)

Type(double click for detail):

Name	Class/Provider	Description
TCP	ocf/heartbeat	ICP resource agent
IPaddr	ocf/heartbeat	Manages virtual IPv4 addresses
IPaddr2	ocf/heartbeat	Manages virtual IPv4 addresses
IPscaddr	ocf/heartbeat	IPscaddr resource agent

② IPaddrを選択

Parameters:

Name	Value	Description
target_role	stopped	press "Default" or "Start" button in toolbar/menu to start the resource
ip	192.168.201.10	IPv4 address

③ パラメータを指定

If belong to a clone or master/slave:

Clone Master/Slave Clone or Master/Slave ID:

clone_max: clone_node_max:

master_max: master_node_max:

④ 追加ボタンを押下

追加(A) キャンセル(C)

Connected to 127.0.0.1

root@node01:/var/lib/ root@node01:~ root@node02:/var/lib/ [root@node02:~] Linux HA Manager

仮想アドレス Operationの追加

The screenshot shows the Linux HA Management Client interface. The main window displays a tree view of resources and nodes. The 'ip01' resource is selected. An 'Add Operation' dialog box is open, showing fields for Name, Description, Interval, Timeout, Start Delay, Disabled, Role, Prereq, and On Fail. The 'Name' field is set to 'start', 'Timeout' is '90s', and 'On Fail' is 'restart'. The 'Add Operation' button is circled in red, and the 'Apply' button is also circled in red. The 'OK' button is also visible.

① Add Operationボタン押下でポップアップ画面表示

② 操作を選択 (start/monitor/stopなど)

③ 値を設定

④ On Fail時の動作を指定

⑤ OKボタン押下でOperationを追加

⑥ start/stop/monitorなど、必要Operationを追加後、Applyボタンを押下

PostgreSQLの追加

The screenshot shows the Linux HA Management Client interface. The main window displays a tree view on the left with 'Resources' expanded to 'group01'. The 'Add Native Resource' dialog box is open, showing the following details:

- Resource ID: pg01
- Belong to group: group01
- Type: postgresql (selected in the list)
- Parameters: target_role: stopped
- Buttons: 追加(A) (Add), キャンセル(C) (Cancel)

Four yellow callouts with numbers 1-4 point to specific elements in the dialog:

- ① リソースIDを指定 (Specify Resource ID)
- ② postgresqlを選択 (Select postgresql)
- ③ パラメータを指定 (Specify parameters)
- ④ 追加ボタンを押下 (Click the Add button)

PostgreSQL Operationの追加

The screenshot shows the Linux HA Management Client interface. The left pane displays a tree view of resources, with 'pg01' (PostgreSQL) selected under 'Resources > group01'. The right pane shows the configuration for 'pg01', including a table of operations.

Name	Description	Interval	Timeout	Start Delay	Disabled	Role	Prereq	On Fa
start			120s					restart
stop			120s					block
monitor		30s	30s	0				restart

At the bottom of the interface, there are buttons for 'Add Operation', 'Delete Operation', 'Apply', and 'Reset'. The 'Add Operation' and 'Apply' buttons are circled in red.

① Add Operationボタン押下でポップアップ画面表示

② start/stop/monitorなど、必要Operationを追加後、Applyボタンを押下

制約 Placesの追加

The screenshot shows the Linux HA Management Client interface. The 'Constraints' section is expanded, and a dialog box titled 'The type of new item' is open. The 'Item Type' dropdown is set to 'place', and the 'OK(O)' button is highlighted. A yellow callout bubble points to the '+' button in the toolbar, and another points to the 'OK(O)' button.

① 追加ボタン押下でポップアップ画面を出す。

② Placeを選択し、OKボタンを押下

制約 Placesの追加

The screenshot shows the Linux HA Management Client interface. The main window displays a tree view of the system configuration. Under 'Constraints', the 'Places' sub-item is selected. An 'Add Place Constraint' dialog box is open, with the following fields:

Field	Value
ID:	place01
Resource:	group01
Score:	100

The 'Resource' field is circled in red, and a yellow callout bubble points to it with the text: ① Resource ID, Resource, Scoreを指定して OK ボタンを押下

The main window also shows a table of system components:

Name	Status
linux-ha	with quorum
Nodes	
node02	running(dc)
node01	running
Resources	
group01	group
ip01	not running
pg01	not running
Constraints	
Places	
Orders	
Colocations	

制約 Places -Expressionの追加

The screenshot shows the Linux HA Management Client interface. The left pane displays a tree view of resources and constraints, with 'place01' selected under 'Constraints' > 'Places'. The right pane shows details for 'place01', including its ID, score, and resource group. A dialog box titled 'Add Expression' is open, allowing the user to define a new expression. The dialog fields are: Attribute: 'hostname', Operation: 'eq', and Value: 'node01'. The 'Add Expression' button in the main panel is circled in red, and a yellow callout box explains that clicking it opens the dialog. Another yellow callout box points to the 'Add Expression' field in the dialog, stating that the attribute, operation, and value should be specified and the OK button pressed. A third yellow callout box points to the 'Add Expression' and 'Apply' buttons in the main panel, stating that clicking 'Apply' after the dialog will reflect the changes.

② Attribute, Operation, Valueを指定し、OKボタンを押下

① Add Expressionボタン押下によりポップアップウィンドウを出す

③ 最後にApplyボタン押下で処理を反映させる。

制約 Placesの追加

The screenshot shows the Linux HA Management Client interface. The left pane displays a tree view of resources and constraints. The right pane shows the configuration for the selected 'place02' constraint.

Attributes:

- ID: place02
- Score: 50
- Resource: group01

Expressions:

Attribute	Operation	Value
#uname	eq	node02

A yellow callout bubble points to the 'Expressions' table with the text: "node02に対するPlaces制約もnode01と同様に設定。Scoreの値は優先的に上げたい方を大きく。"

Buttons at the bottom right: Add Expression, Delete Expression, Apply, Reset.

Bottom status bar: Connected to 127.0.0.1

設定完了

The screenshot shows the Linux HA Management Client interface. The main window is titled "Linux HA Management Client" and has a menu bar with "Connection", "Resources", and "Nodes". Below the menu bar is a toolbar with various icons, including a green circular arrow icon. A yellow callout bubble points to this icon with the text "設定できたら、このボタンを押下。" (When settings are complete, press this button).

The interface is divided into two main panes. The left pane shows a tree view of the configuration:

- linux-ha
 - Nodes
 - node02: running(dc)
 - node01: running
 - Resources
 - group01: group (selected)
 - ip01: not running
 - pg01: not running
 - Constraints
 - Places
 - place01
 - place02
 - Orders
 - Colocations

The right pane shows the details for the selected resource group "group01". It has tabs for "Attributes" and "Parameters". The "Parameters" tab is active, showing a table with the following data:

Name	Value
ordered	true
collocated	true

At the bottom of the right pane, there are buttons for "Add Attribute", "Delete Attribute", "Apply", and "Reset".

The status bar at the bottom of the window shows "Connected to 127.0.0.1" and several terminal windows for different nodes.

リソース開始

The screenshot shows the Linux HA Management Client interface. The main window is titled "Linux HA Management Client" and has a menu bar with "Connection", "Resources", and "Nodes". Below the menu bar is a toolbar with various icons, including a green play button (start) and a green square (stop). The main area is divided into a tree view on the left and a details panel on the right.

The tree view shows a hierarchy of resources:

- linux-ha (with quorum)
 - Nodes
 - node02 (running(dc))
 - node01 (running)
 - Resources
 - group01 (group) - selected
 - ip01 (running on [node01])
 - pg01 (running on [node01])
 - Constraints
 - Places
 - place01
 - place02
 - Orders
 - Colocations

The details panel on the right shows the configuration for the selected resource "group01". It has a tabbed interface with "Attributes" and "Parameters" tabs. The "Parameters" tab is active, showing a table with the following data:

Name	Value
ordered	true
collocated	true

At the bottom of the details panel, there are buttons for "Add Attribute", "Delete Attribute", "Apply", and "Reset".

Two callout boxes provide instructions:

- ① グループのところにカーソルをあわせる (Place the cursor over the group)
- ② startボタンを押下でリソースが起動される。 (Press the start button to start the resource.)

The status bar at the bottom of the window shows "Connected to 127.0.0.1". The taskbar at the very bottom shows several terminal windows and the Linux HA Manager application icon.

疑似故障発生

The screenshot displays the Linux HA Management Client interface. On the left, a tree view shows the configuration for 'linux-ha' with nodes 'node01' and 'node02', and resources 'ip01' and 'pg01'. The 'group01' resource is selected. On the right, a detailed view for 'group01' shows parameters 'ordered true' and 'collocated true'. A terminal window in the foreground shows the output of the 'ifconfig' command for 'eth0:0' on 'node01', displaying network statistics and configuration. A yellow callout bubble points to the terminal with the text: '仮想アドレスを落とすことで擬似的に故障を発生させる。' (Simulate a failure by dropping the virtual address). The terminal prompt shows the command '[root@node01 ~]# ifconfig eth0:0 down'.

Name	Status
linux-ha	with quorum
Nodes	
node02	running(dc)
node01	running
Resources	
group01	group
ip01	running on ['node01']
pg01	running on ['node01']
Constraints	
Places	
place01	
place02	
Orders	
Colocations	

```
root@node01:~  
ファイル(E) 編集(E) 表示(V) 端末(T) タブ(T) ヘルプ(H)  
inet addr:192.168.201.12 Bcast:192.168.201.255 Mask:255.255.255.0  
inet6 addr: fe80::20a:e4ff:fe22:553a/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:1422 errors:0 dropped:0 overruns:0 frame:0  
TX packets:1314 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:100  
RX bytes:346165 (338.0 KiB) TX bytes:313622 (306.2 KiB)  
Base address:0x7000 Memory:d0200000-d0220000  
  
eth0:0 Link encap:Ethernet HWaddr 00:0A:E4:22:55:3A  
inet addr:192.168.201.10 Bcast:192.168.201.138 Mask:255.255.255.0  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
Base address:0x7000 Memory:d0200000-d0220000  
  
lo:0 Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:16384 Metric:1  
RX packets:1697 errors:0 dropped:0 overruns:0 frame:0  
TX packets:1697 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:1858139 (1.7 MiB) TX bytes:1858139 (1.7 MiB)  
[root@node01 ~]# ifconfig eth0:0 down
```

故障検知

The screenshot shows the Linux HA Management Client interface. The main window displays a tree view of resources and nodes. The 'Resources' section is expanded to show 'group01', which contains 'ip01' (failed on [node01]) and 'pg01' (running on [node01]). A yellow callout bubble points to the 'ip01' entry with the text '故障を検知' (Detect failure). The 'Nodes' section shows 'node02' (running(dc)) and 'node01' (running). The 'Attributes' and 'Parameters' tabs are visible, showing 'ordered true' and 'collocated true'.

Terminal window output (root@node01:~):

```
ファイル(E) 編集(E) 表示(V) 端末(T) タブ(T) ヘルプ(H)
inet6 addr: fe80::20a:e4ff:fe22:553a/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:1422 errors:0 dropped:0 overruns:0 frame:0
TX packets:1314 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
RX bytes:346165 (338.0 KiB) TX bytes:313622 (306.2 KiB)
Base address:0x7000 Memory:d0200000-d0220000

eth0:0 Link encap:Ethernet HWaddr 00:0A:E4:22:55:3A
inet addr:192.168.201.10 Bcast:192.168.201.138 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
Base address:0x7000 Memory:d0200000-d0220000

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:1697 errors:0 dropped:0 overruns:0 frame:0
TX packets:1697 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:1858139 (1.7 MiB) TX bytes:1858139 (1.7 MiB)

[root@node01 ~]# ifconfig eth0:0 down
[root@node01 ~]# █
```

フェイルオーバー

The screenshot shows the Linux HA Management Client interface. The main window displays a tree view of resources and nodes. The 'Resources' section is expanded to show 'group01', which contains 'ip01' and 'pg01'. Both are shown as 'running on [node02]'. A yellow callout bubble points to this state with the text: 'リソースがnode01からnode02へフェイルオーバー' (Resource fails over from node01 to node02).

The 'Nodes' section shows 'node01' as 'running' and 'node02' as 'running(dc)'. The 'Constraints' section shows 'Places' with 'place01' and 'place02', 'Orders', and 'Colocations'.

The right-hand pane shows details for 'group01' (ID: group01, Type: group). The 'Attributes' tab is selected, showing 'ordered: true' and 'collocated: true'.

A terminal window is open on 'node01' showing the command 'ifconfig eth0:0 down' being executed. The terminal output shows network details for 'eth0:0' and 'lo'.

```
root@node01:~  
ファイル(E) 編集(E) 表示(V) 端末(T) タブ(T) ヘルプ(H)  
inet6 addr: fe80::20a:e4ff:fe22:553a/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:1422 errors:0 dropped:0 overruns:0 frame:0  
TX packets:1314 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:100  
RX bytes:346165 (338.0 KiB) TX bytes:313622 (306.2 KiB)  
Base address:0x7000 Memory:d0200000-d0220000  
  
eth0:0 Link encap:Ethernet HWaddr 00:0A:E4:22:55:3A  
inet addr:192.168.201.10 Bcast:192.168.201.138 Mask:255.255.255.0  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
Base address:0x7000 Memory:d0200000-d0220000  
  
lo Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:16436 Metric:1  
RX packets:1697 errors:0 dropped:0 overruns:0 frame:0  
TX packets:1697 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:1858139 (1.7 MiB) TX bytes:1858139 (1.7 MiB)  
  
[root@node01 ~]# ifconfig eth0:0 down  
[root@node01 ~]# █
```