

Wake on Lan (WoL)

General information

Since Batocera V33 Beta there is a tool called [ethtool](#) included into Batocera, which basically is a small but powerful advanced network configuration tool. In this case, *ethtool* is being used to enable [Wake on Lan \(WoL\)](#) functionality on your Batocera system, which is not enabled by default in Batocera (yet), so you have to configure it manually. Follow the instructions below to do so.

Requirements

On your Batocera system, the functionality of WoL is dependent on three things that must support WoL:

1. **BIOS:** Your system's BIOS must support WoL and you have to enable WOL within your system's BIOS.
2. **NIC:** Your system's network adapter must support WoL to stay "awake" while the system is powered off.
3. **OS:** You typically need a device driver and/or tool for the specific NIC to be able to enable the hardware's WoL capability. Loading the OS (*Batocera.linux* in this case) usually gives you such a device driver/tool (*ethtool* in this case).

After enabling WoL within the BIOS you can boot your Batocera system, [log in via SSH](#) and check by command line whether your NIC supports WoL or not by executing the following command:

```
ethtool eth0
```

Fortunately, Batocera.linux's underlying network system has [Predictable Network Interface Names](#). Therefore, if you have multiple NICs installed your system you can just use the according ongoing `eth<x>` number of your specific network interface you want to check, e.g. `eth1`, `eth2` and so on, without any hassle. You can list all of them up by executing:

```
ifconfig
```

Now look out for the following two output lines:

Supports Wake-on: → Implies whether and which WoL capabilities your NIC supports (you cannot change those values)

and

Wake-on: → Implies which of them WoL capabilities are enabled for your NIC by the system (*may be enabled by default, you can change those values as shown below*)

Example output:

Supports Wake-on: g

Wake-on: g

The value g means that your NIC supports receiving magic packets and therefore *is* WoL capable. Note that your NIC could also support multiple WoL functions and therefore you would get multiple output values, e.g. Supports Wake-on: pumbg.

Here is a full list of output possibilities (from the `man ethtool` command):

- p → Wake on PHY activity
- u → Wake on unicast messages
- m → Wake on multicast messages
- b → Wake on broadcast messages
- a → Wake on ARP
- g → Wake on MagicPacket™
- s → Enable SecureOn™ password for MagicPacket™
- d → Disable (wake on nothing). This option clears all previous options.

The most important value is the g within **pumbg** which is the minimal WoL capability we need here. Without it being shown, your NIC does not support WoL at all.

Now: If for example your output looks like this:

Supports Wake-on: g

Wake-on:

it means your system's NIC is indeed WoL capable but not yet enabled. To enable the according option persistently you have to create an according [udev rule](#). We will configure this rule to take place whenever a NIC is detected by the system on startup or even on-the-fly, for example if you plugin a USB-LAN adapter into your system while it is running:



Configuring WoL

To do so, create the following file...

```
nano /etc/udev/rules.d/99-wol.rules
```

...and paste the following content (if you want deeper explanation about udev rules, visit [this](#) page):

Note: The code below will enable WoL for all of your system's WoL capable NICs. If for some reason you want to enable WoL only for a specific NIC, replace `eth*` with the according NIC name on the `ACTION==...` code line.

```
ACTION!="add|change", GOTO="end"
```

```
SUBSYSTEM!="net", GOTO="end"
ENV{INTERFACE}!="eth*", GOTO="end"

TEST=="/sys/class/net/${ENV{INTERFACE}}/device/power/wakeup", RUN+="/bin/sh -c
'echo enabled >/sys/class/net/${ENV{INTERFACE}}/device/power/wakeup'"

ACTION=="add|change", SUBSYSTEM=="net", ENV{INTERFACE}=="eth*",
RUN+="/usr/sbin/ethtool -s ${ENV{INTERFACE}} wol g"

LABEL="end"
```

Save the file and quit the editor (Sidenote: No need to make the file executable). Because of [Batocera.linux's overlay based system](#), to make the changes persistent, you have to execute the following command:

```
batocera-save-overlay
```

Now reboot your system.

From now on, when shutting down your system, it can be woken up by every magic packet based WoL client **within the same subnet** by sending a magic packet to the according unique MAC address of your NIC.

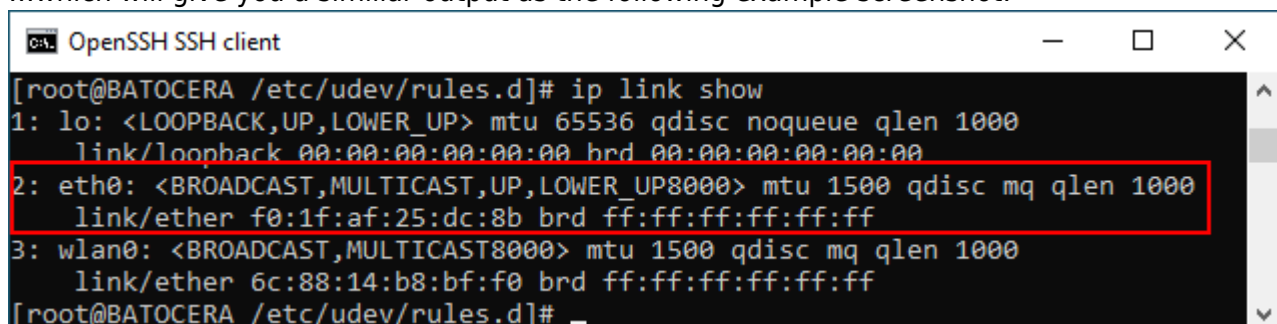
Wake up your Batocera device remotely

Note: It is important to have your WoL client and your Batocera system within the same [Layer 2](#) subnet of your network, otherwise WoL will not work as WoL packages are not going to be routed over different subnets! WoL is a Layer 2 protocol which actually can only be routed with additional and advanced workaround configurations which we are not going to cover here!

To find out the unique MAC address of your Batocera system's NIC, execute the following command on your Batocera system...

```
ip link show
```

...which will give you a similiar output as the following example screenshot:



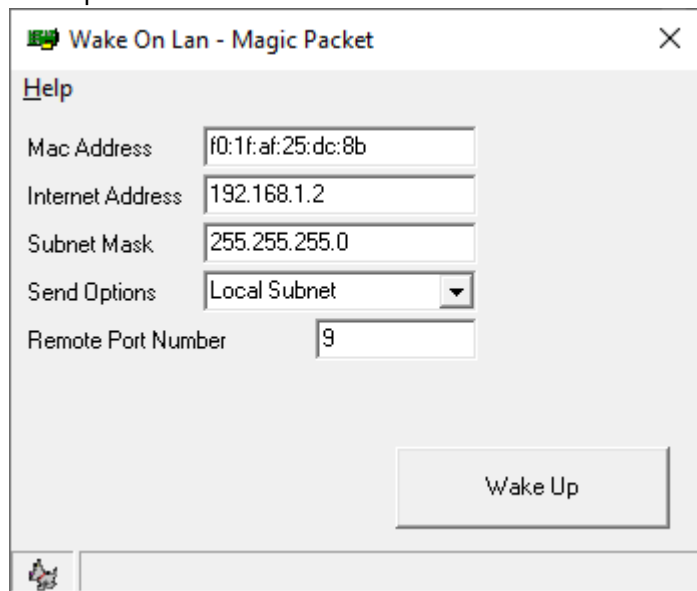
```
OpenSSH SSH client
[root@BATOCERA /etc/udev/rules.d]# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP8000> mtu 1500 qdisc mq qlen 1000
   link/ether f0:1f:af:25:dc:8b brd ff:ff:ff:ff:ff:ff
3: wlan0: <BROADCAST,MULTICAST8000> mtu 1500 qdisc mq qlen 1000
   link/ether 6c:88:14:b8:bf:f0 brd ff:ff:ff:ff:ff:ff
[root@BATOCERA /etc/udev/rules.d]#
```

As you can see on the screenshot above, the MAC address for the NIC eth0 is f0:1f:af:25:dc:8b.

Note: A MAC address is always based on twelve characters (48 bit) where every character pair is divided to its next character pair by a : or a - symbol.

You can now use the given MAC address to wake your Batocera system up remotely from another device (e.g. PC, Smartphone, ...) which is **within the same Layer 2 subnet** and has a WoL client (GUI based or command line based WoL client) installed.

Example of a basic GUI based WoL client for Windows (which you can download [here](#)):



- **Mac Address** → MAC address of your NIC on your Batocera system you want to wake up
- **Internet Address** → Insert the IP address of your according Batocera system's NIC (If you only have one NIC on your WoL client, you can leave this empty)
- **Subnet Mask** → [Netmask](#) of your network's subnet
- **Send options** → Choose Local Subnet
- **Remote Port Number** → Choose 9 (default) or 7

Click on Wake Up to wake your Batocera system up.

Troubleshooting

If you have followed the instructions on this page precisely and still are not able to wake your device up, try to play around with your WoL client its settings (e.g. change **Remote Port Number** from 7 to 9 or vice versa. Or, in the example above, you could try to change **Send Options** from Local Subnet to Internet which sometimes helps.).

From:
<https://www.wiki.batocera.org/> - **Batocera.linux - Wiki**

Permanent link:
https://www.wiki.batocera.org/wake_on_lan?rev=1642095354

Last update: **2022/01/13 17:35**

