

ifup and ifdown

Apr 2020

Overview

If have been using **ifup** and **ifdown** for quite some time now, and as far as i was aware the syntax was pretty simple, however I got really caught out the other day and wanted to share what I discovered (not entirely on my own, with the help of some other, way more clever people).

Simple Syntax

So, to take down an interface, I would use the following:

```
ifdown eth1
```

Device 'eth1' successfully disconnected.

Simple, and to bring that interface back up:

```
ifup eth1
```

Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/3)

Wow, too easy. I use this normally when I have changed and IP Address and want to invoke that change quickly (without using the systemctl command or a reboot).

What Went Wrong?

Now, this is not going to affect everyone, but in our systems we sometimes have to prove redundancy by taking down an interface, letting a device detect this and watching the system automatically switch. Sounds simple enough, but it seems that using a simple **ifdown** or **ifup** does not cause a redundancy switch, you need to do a bit more.

To properly take the interface down, so that it is recognised as being down from a 'link'perspective, you must use the following syntax:

```
ip link set eth1 down
```

(obviously use your interface name, not mine)

And inversly:

```
ip link set eth1 up
```

This will ensure that the interface goes fully down, and that other devices can recognize that the interface is down.

I sincerely hope this saves you some pain :)

From:
<http://cameraangle.co.uk/> - WalkerWiki - wiki.alanwalker.uk

Permanent link:
http://cameraangle.co.uk/doku.php?id=wiki:ifup_and_ifdown

Last update: **2023/08/17 17:37**

