# Firewalling virtuals with ebtables

I had a situation where I wanted to control access from one virtual to the others on the network. It could have been done via NAT, but the eventual goal is to have several virtual machines which can not “see” each other, and did now want to go building several virtual networks. So, I researched ebtables.

ebtables is a network filtering tool designed to work with Unix Bridges. Most (all?) virtualization software supports, and even recommends, using a network bridge.

Since a bridge forwards, the filtering is done under the FORWARD rule.

In this example, we are looking at three machines:

| Name | MAC | Description |
|---|---|---|
| Win10 | 00:16:3e:6b:26:70 | this is the machine we want to restrict |
| router | 00:16:3e:bd:26:71 | this is the router |
| manage | 00:16:3e:37:26:72 | this is the one internal machine which may be reached |

```
# first, flush all tables (restore to default)
ebtables -F
# let Win10 talk to router
ebtables -A FORWARD -s 00:16:3e:6b:26:70 -d 00:16:3e:bd:26:71 -j ACCEPT
# let router talk to Win10
ebtables -A FORWARD -s 00:16:3e:bd:26:71 -d 00:16:3e:6b:26:70 -j ACCEPT
# let Win10 talk to manage
ebtables -A FORWARD -s 00:16:3e:6b:26:70 -d 00:16:3e:37:26:72 -j ACCEPT
# let manage talk to Win10
ebtables -A FORWARD -s 00:16:3e:37:26:72 -d 00:16:3e:6b:26:70 -j ACCEPT
# not sure why, but we need these two protocols usable
ebtables -A FORWARD -s 00:16:3e:6b:26:70 -p 800 -j ACCEPT
ebtables -A FORWARD -s 00:16:3e:6b:26:70 -p 806 -j ACCEPT
# Drop all other traffic where Win10 is the source
ebtables -A FORWARD -s 00:16:3e:6b:26:70 -j DROP --log
# and drop all other traffic where Win10 is the destination
ebtables -A FORWARD -d 00:16:3e:6b:26:70 -j DROP --log
# show the user what the tables look like.
ebtables -L
```