

TCP optimizations

Generic sysctl TCP optimizations

Enable BBR kernel module

```
$ sudo modprobe tcp_bbr
$ echo "tcp_bbr" | sudo tee /etc/modules-load.d/
bbr.conf
```

Apply changes

sudo vim /etc/sysctl.d/99-network-tuning.conf and paste these in:

```
# These were taken from:
# - https://github.com/zchee/docker-h2o/blob/master/
  sysctl.conf
# - https://hpbn.co/building-blocks-of-tcp/
```

```
# Turn on the tcp_window_scaling
net.ipv4.tcp_window_scaling = 1
```

```
# Increase the write-buffer-space allocatable
net.ipv4.tcp_wmem = 8192 65536 16777216
net.ipv4.udp_wmem_min = 16384
net.core.wmem_default = 131072
net.core.wmem_max = 16777216
```

```
# Increase the read-buffer-space allocatable
net.ipv4.tcp_rmem = 8192 65536 16777216
net.ipv4.udp_rmem_min = 16384
net.core.rmem_default = 131072
net.core.rmem_max = 16777216
```

```
# Allow the TCP fastopen flag to be used, beware
some firewalls do not like TF0! (kernel > 3.7)
net.ipv4.tcp_fastopen=3
```

```
# Avoid falling back to slow start after a
connection goes idle
# keeps our cwnd large with the keep alive
connections (kernel > 3.6)
net.ipv4.tcp_slow_start_after_idle=0
```

```
# Connection backlog (important for servers under
load)
net.core.somaxconn = 65535
net.core.netdev_max_backlog = 65535
net.ipv4.tcp_max_syn_backlog = 65535

# Ephemeral port range (if you make many outbound
connections)
net.ipv4.ip_local_port_range = 1024 65535

# SYN flood protection
net.ipv4.tcp_syncookies = 1

# BBR generally outperforms CUBIC, at least on lossy
or high-latency links
net.core.default_qdisc = fq
net.ipv4.tcp_congestion_control = bbr

# More generous file descriptors
fs.file-max=2097152
fs.nr_open=2097152
```

Confirm changes

```
$ sudo sysctl --system
```

Configure TCP Initial Congestion Window

Network interface can be found by running:

```
$ ip route | awk '/default/ {print $5}'
enp0s6
```

Confirm the metadata destination route is correct and exists
(**169.254.0.0/16** in this case):

```
$ ip route
default via 10.0.0.1 dev enp0s6 proto dhcp src
10.0.0.28 metric 100
10.0.0.0/24 dev enp0s6 proto kernel scope link src
10.0.0.28 metric 100
10.0.0.1 dev enp0s6 proto dhcp scope link src
10.0.0.28 metric 100
```

```
169.254.0.0/16 dev enp0s6 proto dhcp scope link src
10.0.0.28 metric 100
169.254.169.254 dev enp0s6 proto dhcp scope link src
10.0.0.28 metric 100
```

Apply changes

Run **sudo vim /etc/systemd/network/05-enp0s6.network** and add these in (base values are from **/run/systemd/network/10-netplan-enp0s6.network**):

```
[Match]
```

```
Name=enp0s6
```

```
[Link]
```

```
MTUBytes=9000
```

```
[Network]
```

```
DHCP=ipv4
```

```
LinkLocalAddressing=ipv6
```

```
[DHCP]
```

```
RouteMetric=100
```

```
UseMTU=true
```

```
UseRoutes=false
```

```
UseGateway=false
```

```
[Route]
```

```
Gateway=10.0.0.1
```

```
InitialCongestionWindow=15
```

```
InitialAdvertisedReceiveWindow=15
```

```
[Route]
```

```
Destination=169.254.0.0/16
```

```
Scope=link
```

Confirm changes

And then run:

```
$ sudo networkctl reload
```

```
# or sudo systemctl restart systemd-networkd
```

```
# Now you should see:
```

```
$ ip route show
default via 10.0.0.1 dev enp0s6 proto static onlink
initcwnd 15 initrwnd 15
10.0.0.0/24 dev enp0s6 proto kernel scope link src
10.0.0.28 metric 100
169.254.0.0/16 dev enp0s6 proto static scope link
169.254.169.254 dev enp0s6 proto dhcp scope link src
10.0.0.28 metric 100
```