Device Setup

- Justine
- Hifumi: Printer server
- Ann: Minecraft Server
- Igor
- Igor: Zabbix
- Ann: Kodi
- OVH VPS

Justine

Interfaces

enp1s0

- HOME VLAN; untagged

```
ip link set enp1s0 up
ip addr replace 192.168.1.2/24 dev enp1s0
ip route add default via 192.168.1.1 dev enp1s0
```

mgmt@enp1s0

- MGMT VLAN; tagged VLAN 100

```
ip link add link enp1s0 name mgmt type vlan id 100
ip link set mgmt up
ip addr replace 192.168.100.2/24 dev mgmt
```

docker0

- 172.18.0.1/16

Set up automatically by docker.

Docker namespaces use virtual interface that gets bridged with docker0.

Routing

Forwarding

Enabled but packets dropped by default on firewall.

```
sysctl net.ipv4.ip_forward=1
iptables -P FORWARD DROP
```

Mullvad

Mullvad VPN outgoing traffic is MASQUERADEed for it to get Mullvad assigned internal IP.

```
# Mullvad gateway
iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -o mullvad -j MASQUERADE
iptables -t nat -A POSTROUTING -s 172.17.1.0/24 -o mullvad -j MASQUERADE
```

When Mullvad VPN is up/down additional firewall rules are added:

```
PostUp = iptables -A FORWARD -o mullvad -i enp1s0 -j ACCEPT && iptables -A FORWARD -i mullvad
-o enp1s0 -j ACCEPT
PreDown = iptables -D FORWARD -o mullvad -i enp1s0 -j ACCEPT && iptables -D FORWARD -i mullvad
-o enp1s0 -j ACCEPT
```

This will allow forwarding between mullvad (VPN) and enp1s0 (HOME) networks.

Vpn

When this WireGuard endpoint is enabled additional rules are added:

```
PostUp = iptables -A FORWARD -o vpn -i enp1s0 -j ACCEPT && iptables -A FORWARD -i vpn -o
enp1s0 -j ACCEPT && iptables -A FORWARD -o vpn -i mullvad -j ACCEPT && iptables -A FORWARD -i
vpn -o mullvad -j ACCEPT
PreDown = iptables -D FORWARD -o vpn -i enp1s0 -j ACCEPT && iptables -D FORWARD -i vpn -o
enp1s0 -j ACCEPT && iptables -D FORWARD -o vpn -i mullvad -j ACCEPT && iptables -D FORWARD -i
vpn -o mullvad -j ACCEPT
```

This will allow:

1. vpn users to access local network (HOME),
2. vpn users to access the internet via mullvad VPN interface.

Docker

Allow traffic from Docker (IPHole) to be originating from justine IP if routed through default HOME VLAN gateway (caroline) - this is when VPN is turned off to keep DNS working.

```
# VPN gateway (used if mullvad is stopped)
iptables -t nat -A POSTROUTING -s 172.17.1.1/24 -o enp1s0 -j MASQUERADE
```

> PIHole uses Mullvad's hosted DNS server at: 193.138.218.74. It is accessible over VPN and also without it.

> Any DNS port 53 packet going over Mullvad VPN will be SNAT'ed to Mullvads DNS server transparently to prevent DNS leaks. This means that running DNS resolved (unbind) makes no sense since all DNS requests will end up on Mullvad's server anyway.

Local networks

Allow access to other local networks via caroline:

```
ip route add 192.168.1.0/16 dev enp1s0 via 192.168.1.1
```

VPN

Outpost

- caroline UDP port: `34564`
- justine UDP port: `51822`

Used for devices to connect in to Justine (no forwarding is set up currently).

vpn

- caroline UDP port: `34563`
- justine UDP port: `51821`

For all devices to VPN-in to the G/W from internal networks and also from the internet.

VPN access from outside the network

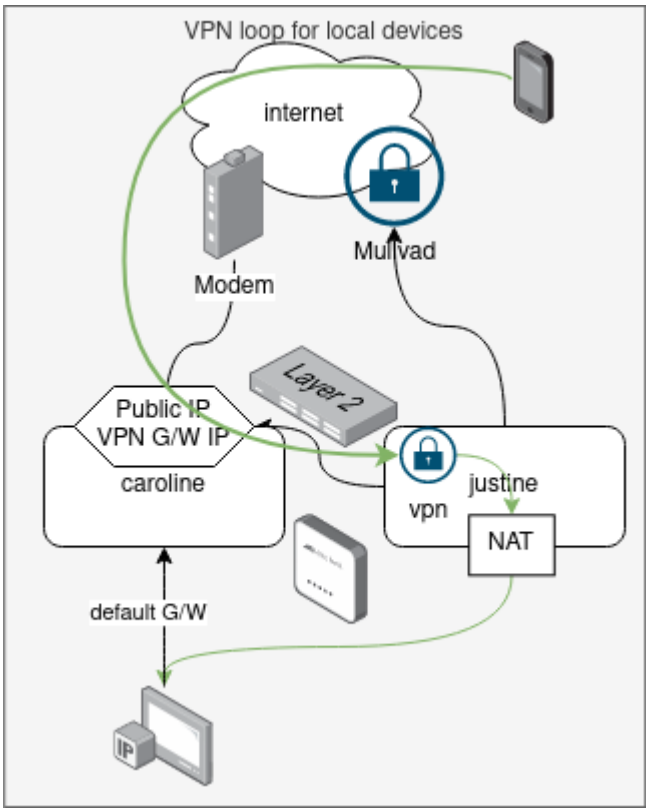
> DEPRECATED: This is no longer the case as I don't have ability to forward IPv4 ports into the network or set ISP router in bridge mode.

> TODO: Document how VPN connection is established from Justine to Vps and there incoming VPN connections are forwarded back to Justine. Justine to not route this connection to Vps via mullvad...

```
iptables -t nat -A POSTROUTING -s 172.17.1.0/24 -o enp1s0 -j MASQUERADE
```

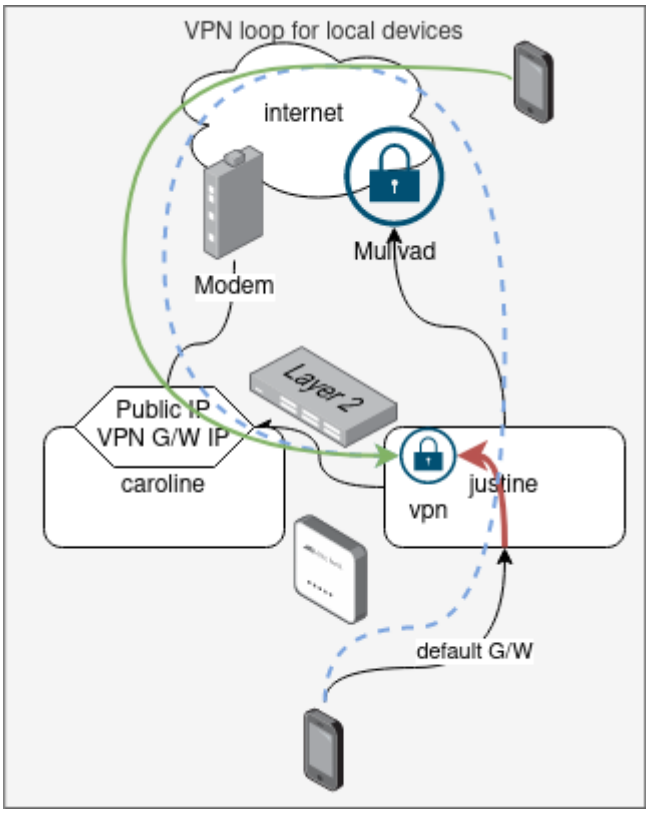
Traffic from VPN (`172.17.1.0/24`) needs to be MASQUERADE'ed when going out to internal network because there are devices configured with **caroline** as default G/W. Also **justine**, when not connected to Mullvad will use **caroline** as default G/W.

This means that all traffic from external devices will looks like coming from **caroline**.



VPN access from withing the network

Devices like laptop or phone will be on always-on home VPN. This means that they will be connecting to VPN via public IP to reach justine.



This entry will capture attempt from devices that route via justine (default G/W `192.168.1.2`) to justine to prevent traffic going out to Mullvad and comming back to caroline and down to justine.

```
iptables -t nat -A PREROUTING -s 192.168.1.0/16 -d 57.128.183.232 -p udp --dport 34563 -j DNAT
--to-destination 192.168.1.2:51821
```

> The /16 prefix is used so this rule captures all internal subnets.

> Public IP in the rule will need to be updated if it ever changes! This IP is the IP of VPN endpoint - caroline public DHCP assigned IP/Virgin Media IP.

> MYSTERY
> - this only gets few packet hit, so bulk traffic is bypassing this rule
> - when connected to MGMT with laptop the traffic to HOME network is slow, looks like it is going through the loop

Hifumi: Printer server

Hardware - R2S

Network

- `eth0` - RTL8211E
- `eth1` - RTL8153

The RTL8153 device needs MAC assignment as it will use random value by default:

/etc/udev/rules.d/10-network-mac-addr.rules

```
SUBSYSTEM=="net", ACTION=="add", KERNEL=="eth1", PROGRAM="/sbin/ip link set %k address
8a:f4:c8:41:48:35"
```

Leds

Make red sys led bright when we have booted to runit:

/etc/runit/core-services/03-_led.sh

```
echo "1" > /sys/class/leds/nanopi-r2s:red:sys/brightness
```

Make red sys to blink on SD card activity and wan/lan on data transfers between eth0 and eth1 (GUEST/internet network access):

/etc/rc.local

```
modprobe ledtrig-netdev
echo "netdev" > /sys/class/leds/nanopi-r2s:green:lan/trigger
echo "eth0" > /sys/class/leds/nanopi-r2s:green:lan/device_name
echo "1" > /sys/class/leds/nanopi-r2s:green:lan/link
```

```
echo "1" > /sys/class/leds/nanopi-r2s:green:lan/tx
echo "0" > /sys/class/leds/nanopi-r2s:green:lan/rx
echo "netdev" > /sys/class/leds/nanopi-r2s:green:wan/trigger
echo "eth1" > /sys/class/leds/nanopi-r2s:green:wan/device_name
echo "1" > /sys/class/leds/nanopi-r2s:green:wan/link
echo "1" > /sys/class/leds/nanopi-r2s:green:wan/tx
echo "0" > /sys/class/leds/nanopi-r2s:green:wan/rx
echo "mmc0" > /sys/class/leds/nanopi-r2s:red:sys/trigger
```

Printer setup

Configure and unpause all printers

```
#!/bin/sh -x
lpstat -le | grep ' permanent ' | cut -f1 -d' ' | while read P; do
□lpadmin -p "$P" -o printer-error-policy=retry-current-job
□lpadmin -p "$P" -o printer-is-shared=true
□lpadmin -p "$P" -E
done
```

Printer status

```
lpstat -t
```

Guest VLAN bridge

This will bride `eth0` to GUEST VLAN (10) on `eth1`.

in `/etc/rc.local`:

```
ip link set eth0 addrgenmode none up
ip link add link eth1 name guest type vlan id 10
ip link add br-guest type bridge
ip link set guest master br-guest
ip link set eth0 master br-guest addrgenmode none
```

```
ip link set br-guest addrgenmode none up
```

Prevent DHCP from running on `eth0`.

In `/etc/sv/dhcpcd/conf`:

```
OPTS="-M --denyinterfaces eth0"
```

Ann: Minecraft Server

Server setup

Java install:

```
xi openjdk21-jre
xbps-alternatives -s openjdk21-jre
```

Generic fabric server

Download correct server JAR from: https://fabricmc.net/use/server/

Run script (fix the jar file name):

```
#!/bin/sh
exec java -Xmx4G -jar fabric-server-mc.1.20.1-loader.0.14.22-launcher.0.11.2.jar nogui
```

EULA file `eula.txt`:

```
#By changing the setting below to TRUE you are indicating your agreement to our EULA
(https://account.mojang.com/documents/minecraft_eula).
#Mon May 02 18:35:52 IST 2022
eula=true
```

Server settings `server.properties`:

```
#Minecraft server properties
#Sat Jun 01 12:08:44 IST 2024
enable-jmx-monitoring=false
level-seed=xxx
rcon.port=25575
enable-command-block=false
gamemode=survival
enable-query=false
generator-settings={}
```

```
enforce-secure-profile=true
level-name=HxD
motd=HxD Mods
query.port=25565
pvp=false
generate-structures=true
max-chained-neighbor-updates=1000000
difficulty=normal
network-compression-threshold=256
require-resource-pack=false
max-tick-time=60000
max-players=20
use-native-transport=true
enable-status=true
online-mode=true
allow-flight=true
initial-disabled-packs=
broadcast-rcon-to-ops=true
view-distance=12
resource-pack-prompt=
server-ip=
allow-nether=true
server-port=25565
enable-rcon=true
sync-chunk-writes=true
op-permission-level=4
prevent-proxy-connections=false
hide-online-players=false
resource-pack=
entity-broadcast-range-percentage=100
simulation-distance=10
player-idle-timeout=0
rcon.password=xxx
force-gamemode=false
rate-limit=0
hardcore=false
white-list=false
broadcast-console-to-ops=true
spawn-npcs=true
```

```
previews-chat=false
spawn-animals=true
function-permission-level=2
initial-enabled-packs=vanilla,fabric
level-type=minecraft\:normal
text-filtering-config=
spawn-monsters=true
enforce-whitelist=false
resource-pack-sha1=
spawn-protection=16
max-world-size=29999984
```

Backups

Backup script :

```
#!/bin/fish
install -d backup
set LEVEL (cat server.properties | grep '^level-name=' | cut -d= -f2)
tar cv "$LEVEL" | zstd > "backup/"(date -Ins)"-$LEVEL.tar.zstd"
ls -tr backup/*-$LEVEL.tar.zstd | head -n -10 | while read F
  rm -v "$F"
end
```

Master backup script - assuming server games are in `games` directory and there is a `minecraft` runit service set up:

```
#!/bin/sh
sudo sv stop minecraft
sudo sv stop minecraft || exit 1
sync
cd games/`ls games/*/logs/latest.log -t | head -n1 | awk -F '/' '{print $2}'`/ && ./backup.sh
echo syncing...
sync
```

Autostart Minecraft client

```
#!/bin/sh
sleep 2
notify-send -u normal -a autostart -r 99001 "Waiting for controller..."
while ! bluetoothctl info AC:FD:93:98:FE:F7 | grep -q 'Connected: yes'; do echo -n "."; sleep
1; done
notify-send -u normal -a autostart -r 99001 "Waiting for network..."
while ! ping -c 1 -q microsoft.com >/dev/null; do echo -n 'x'; sleep 1; done
notify-send -u low -a autostart -r 99001 "Running Minecraft"
cd bin/MultiMC && ./run.sh
```

```
#!/bin/sh
exec ./MultiMC --launch 'HxD Mods III' --server localhost:25565 --profile $PROFILE_NAME
```

Using rcon

XBPS template:

```
# Template file for 'mcrcon'
pkgname=mcrcon
version=0.7.2
revision=0
build_style=gnu-makefile
short_desc="Console based Minecraft rcon client for remote administration and server
maintenance scripts"
maintainer="Orphaned <orphan@voidlinux.org>"
license="Zlib"
homepage="https://sourceforge.net/projects/mcrcon/"
distfiles="https://github.com/Tiiffi/mcrcon/archive/refs/tags/v${version}.tar.gz"
checksum=1743b25a2d031b774e805f4011cb7d92010cb866e3b892f5dfc5b42080973270
```

Install: `xi mcrcon`

```
#!/bin/sh
MCRCON_HOST=localhost MCRCON_PORT=25575 MCRCON_PASS=xxxxx mcrcon
```

Allow-listing players

```
whitelist add player123
whitelist list
```

VM setup

- IP: `192.168.50.152`
- port: `25565` rcon: `25575`

> Needed to set MTU to 1400 to fix authentication issues with MS server

In `/etc/rc.local`:

```
# curl -v https://13.107.246.52 --insecure
ip link set eth0 mtu 1400
```

Igor

Network

After reboot need to add default route manually for Igor to find access to internet.

```
ip route add default via 192.168.100.1
```

Backups

Local

They run to `/var/lib/vz/dump` which is the root volume that has 94GB in total so only keep 2 backups max.

Umma over SMB

Backups go to SMB mount at `/mnt/pve/umma` on Igor that mounts `Igor` share from Umma.

Failing backups

Looks like compression is not on the fly but it first dumps data uncompressed and then runs compression which fails on SMB mount... https://community.nethserver.org/t/proxmox-help-needed-proxmox-backup-ends-with-broken-pipe/18537/2

Can set where the "temp" file is created so it goes to local drive first:

/etc/vzdump.conf

```
tmpdir: /var/lib/vz/dump/temp
```

This is not true for VM backups... I see files with `.dat` created during backup where size matches compressed size and "tmpdir" is not used much.

Looks like the problem is on CIFS level:

```
[9092263.225010] CIFS: Attempting to mount \\192.168.0.8\Igor
[9092263.385599] CIFS: VFS: parse_server_interfaces: malformed interface info
[9092318.751704] CIFS: VFS: \\192.168.0.8 Error -104 sending data on socket to server
[9092318.925726] CIFS: reconnect tcon failed rc = -11
[9092329.032446] CIFS: VFS: \\192.168.0.8 Error -32 sending data on socket to server
[9092338.103933] CIFS: VFS: \\192.168.0.8 Error -104 sending data on socket to server
[9092348.628780] CIFS: VFS: \\192.168.0.8 Error -32 sending data on socket to server
[9092358.347996] CIFS: VFS: \\192.168.0.8 Error -104 sending data on socket to server
[9092368.365350] CIFS: VFS: \\192.168.0.8 Error -32 sending data on socket to server
[9092368.367144] CIFS: VFS: \\192.168.0.8 Error -512 sending data on socket to server
[9092378.738559] CIFS: VFS: \\192.168.0.8 Error -104 sending data on socket to server
[9092379.353289] CIFS: VFS: No writable handle in writepages rc=-11
...
[9092379.387539] CIFS: VFS: No writable handle in writepages rc=-9[9092768.976150] CIFS: VFS:
\\192.168.0.8 Error -32 sending data on socket to server
[9092788.488714] CIFS: VFS: \\192.168.0.8 Error -104 sending data on socket to server
[9092788.936658] CIFS: VFS: \\192.168.0.8 Error -104 sending data on socket to server
...
```

On Umma I have checked:

- Disable multiple connection from the same IP address
- Enable asynchronous read (auto select for next option)
- Enable SMB3 Multichannel

And now I don't get `parse_server_interfaces: malformed interface info` and full backup worked with no errors.

Umma disks can't hibernate

Every 10 seconds Igor probes disk causing Umma to log:

```
User [igor] from [IGOR(192.168.0.50)] via [CIFS(SMB3)] accessed shared folder [Igor].
```

This is known issue https://forum.proxmox.com/threads/high-rate-access-the-smb-shared-folder.140759/ and complete disregard from ProxMox devs :/

The probing stops when storage is disable, and starts when re-enabled.

```
pvesm set umma --disable true
pvesm set umma --disable false
```

I have set up crontab jobs to enable/disable storage around backup time `crontab -e`:

```
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

55 1 * * mon pvesm set umma --disable false
0 4 * * mon pvesm set umma --disable true
```

Alternatively disable connection checking (require a reboot/pve services restart):

> Make sure you can access ProxMox via SSH. If you make a mistake PVE services may fail to start and there won't be UI available.

```
--- /usr/share/perl5/PVE/Storage/CIFSPlugin.pm.orig	2024-02-26 17:50:07.546260476 +0000
+++ /usr/share/perl5/PVE/Storage/CIFSPlugin.pm	2024-02-26 18:06:23.136144956 +0000
@@ -254,6 +254,7 @@
     }
 }

+=begin
 sub check_connection {
     my ($class, $storeid, $scfg) = @_;

@@ -287,6 +288,7 @@

     return 1;
 }
+=cut

 # FIXME remove on the next APIAGE reset.
 # Deprecated, use get_volume_attribute instead.
```

Igor: Zabbix

Templates

- Synology DiskStation SNMPv3 (modified): Synology DiskStation SNMPv3.yaml
- Switch Interfaces SNMPv2: Switch Interfaces SNMPv2.yaml

Ann: Kodi

SSL certificate for web UI

1. Put certificate in `/storage/.kodi/userdata/server.pem` (*pem*) and key in `/storage/.kodi/userdata/server.key` (unencrypted; *pem*)
2. Go to **Settings** / **Services** / **Control** and turn on **Enable SSL**.

OVH VPS

Since I no longer have public IPv4 assigned to my modem/router and no way to enable modem/bridge mode I use VPS to terminate incoming traffic for HTTP and VPN-in.

- Type: VPS `vps2020-starter-1-2-20` (1 vcore, 2 GiB RAM, 20 GB HDD)
- Location: London `os-uk2`
- OS: AlmaLinux 9
- IPv4: `57.128.183.232`
- IPv6: `fe80::f816:3eff:fe78:d4a7/64`

Justine (`172.17.100.3`) and Server-gw (`172.17.100.2`) establish VPN connection to it on port `51322` using `172.17.100.1/24` VPS.

```
[Interface]
PrivateKey = <REDACTED>
MTU = 1380
ListenPort = 51322
Address = 172.17.100.1/24

[Peer]
PublicKey = PTu13g5XRIVt+i1DL3g5QujHwL6TJaHkC9z8Kw7pwQE=
AllowedIPs = 172.17.100.2/32
PersistentKeepalive = 300

[Peer]
PublicKey = EnRj9UgoE1qyQ9qK9OU3jZ39tpAo24FTZMdT6nQN0wY=
AllowedIPs = 172.17.100.3/32
PersistentKeepalive = 300
```

IP tables configuration is used to forward packets to Justine and Server-gw:

```
iptables -P INPUT DROP
iptables -A INPUT ! -i vps -d 172.17.100.0/24 -j DROP
iptables -A INPUT -p icmp -j ACCEPT
iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A INPUT -m tcp -p tcp --dport 22 -m recent --rcheck --seconds 30 --name SSH -j
ACCEPT
iptables -A INPUT -m tcp -p tcp --dport <REDACTED> -m recent --set --name SSH -j DROP
```

```
iptables -A INPUT -p udp -m udp --dport 51322 -j ACCEPT

sysctl net.ipv4.ip_forward=1
iptables -P FORWARD DROP
iptables -A FORWARD -o vps -i eth0 -j ACCEPT
iptables -A FORWARD -i vps -o eth0 -j ACCEPT

iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
iptables -t nat -A POSTROUTING -o vps -d 172.17.100.3 -j MASQUERADE
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to-destination
172.17.100.2:80
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j DNAT --to-destination
172.17.100.2:443
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 4443 -j DNAT --to-destination
172.17.100.2:4443
iptables -t nat -A PREROUTING -i eth0 -p udp --dport 10000 -j DNAT --to-destination
172.17.100.2:10000
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 25565 -j DNAT --to-destination
172.17.100.2:25565
iptables -t nat -A PREROUTING -i eth0 -p udp --dport 34563 -j DNAT --to-destination
172.17.100.3:51821
iptables -t nat -A PREROUTING -i eth0 -p udp --dport 34564 -j DNAT --to-destination
172.17.100.3:51822
```

Vps acts as VPN router for all traffic for Server-gw and by extension for all SERVER VLAN hosts. This way enter and exit IP for servers is the public IP of Vps. Servers can also see original client IP as it is not NATed on the way in.

For VPN-in Vps will NAT connections to Justine since Justine uses Mullvad or ISP IP as default G/W.

Blocking forwarded traffic

```
ipset create forward-drop hash:net
iptables -I FORWARD 1 -m set --match-set forward-drop src -j DROP
```

List IP and add IP to block list:

```
ipset list
ipset add forward-drop 66.249.0.0/16
```