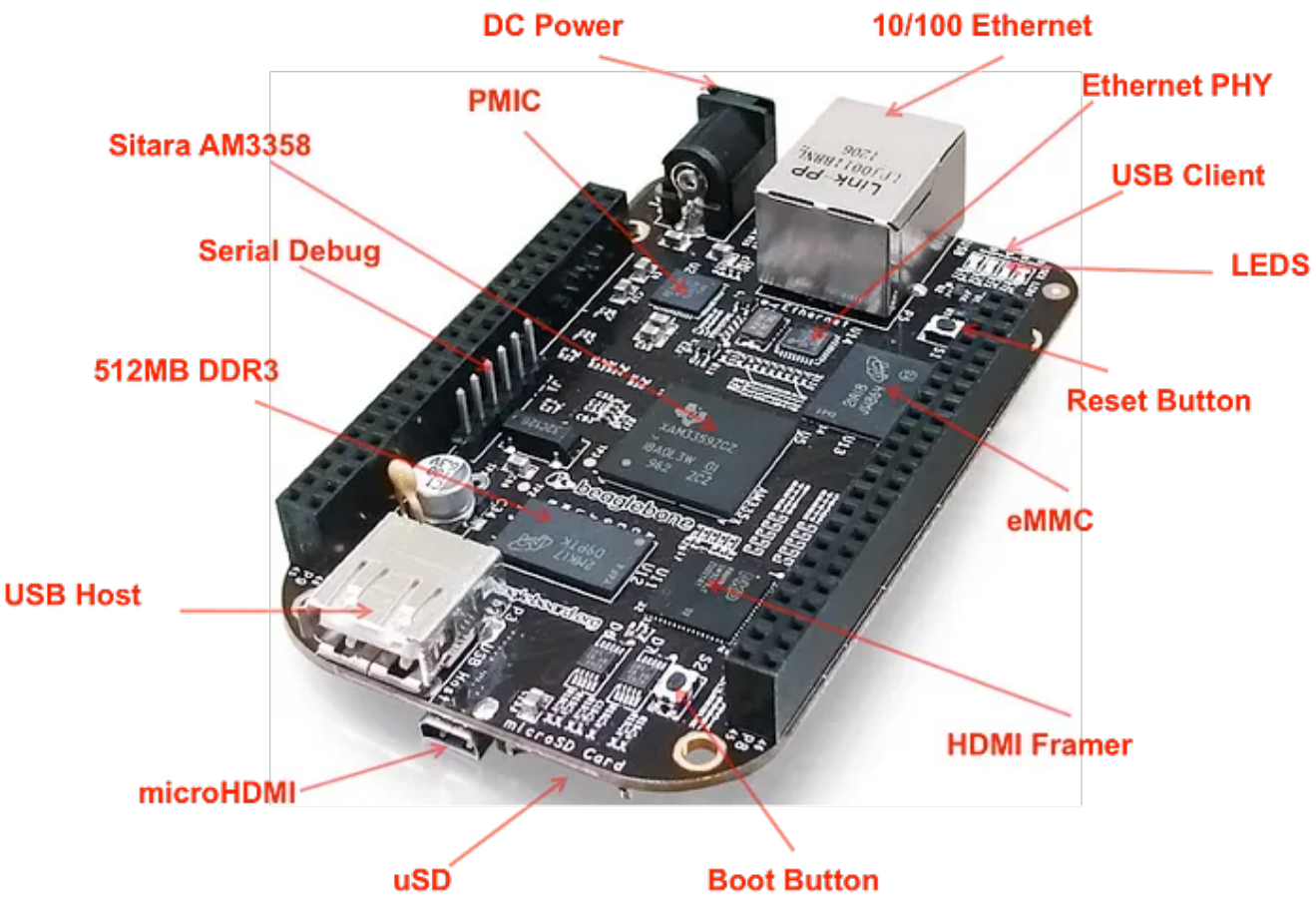
BeagleBone

IoT Beaglebone black
Purpose: ~~host Unify Controller and maybe other services~~ DNS 2
This IoT is delivered with super ugly images containing tons of BS™ software like cloud9 that takes tons of MB to then make you spend hours removing/reconfiguring them.
Hardware: good
OS image: catastrophic

INSTALL Debian

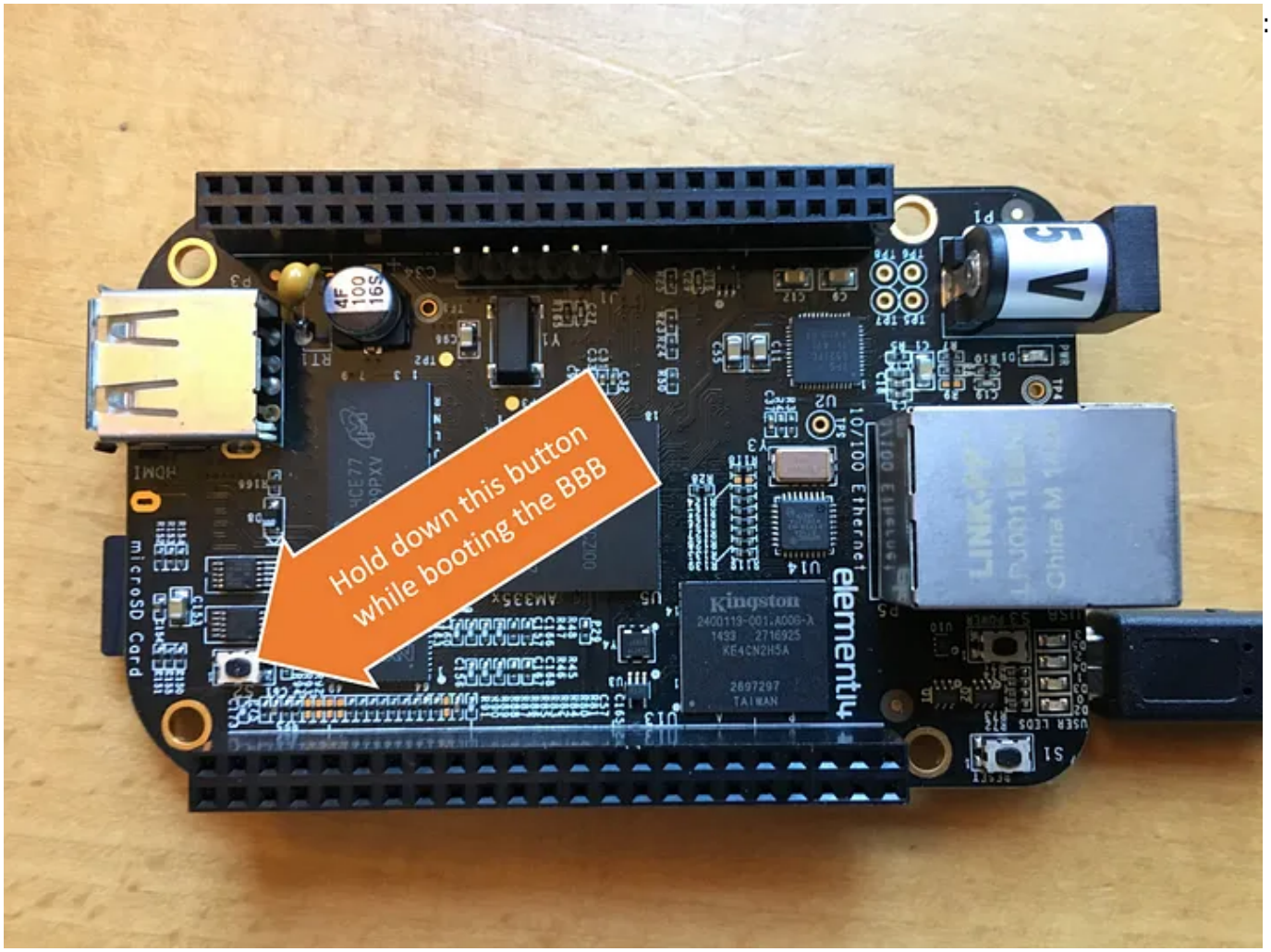
The ISOs are full of node.js crap website that needs extreme hacking for Pi-hole to work !…
→ https://elinux.org/Beagleboard:BeagleBoneBlack_Debian
→ https://learn.adafruit.com/beaglebone-black-installing-operating-systems?view=all

→ https://beagleboard.org/latest-images/ - take the one that flashes eMMC



Then

:



Remove the crap

```
ssh debian@IP (pass = temppwd)
sudo su
pwd
vi /etc/ssh/sshd_config # inet & PermitRootLogin
systemctl restart sshd
```

```
apt update
apt upgrade -y
reboot
```

Debian 10:

```
systemctl stop cloud9.service
systemctl stop cloud9.socket
systemctl disable cloud9.service
systemctl disable cloud9.socket

apt remove --purge nginx*
apt remove --purge c9-core-installer nodejs* apache2*
rm -rf /usr/local/lib/node_modules/bonescript
```

```
apt autoremove
apt autoclean
rm -rf /opt/*
reboot
```

static IP / remove connman - Debian 10:!:

```
vi /etc/connman/main.conf
#NetworkinterfaceBlacklist=eth0,SoftAp0,usb0,usb1
```

vi /etc/network/interfaces ... </code> # The primary network interface

```
 vi /etc/network/interfaces
```

```
auto eth0
iface eth0 inet static
  address 192.168.1.11
  netmask 255.255.255.0
  gateway 192.168.1.1
  dns-nameservers 192.168.1.10 192.168.1.11
...
```

```
systemctl disable connman
reboot
```

```
apt remove --purge connman
systemctl disable dnsmasq
apt remove --purge dnsmasq
rm -rf /etc/resolvconf /etc/dnsmasq.d
```

```
vi /etc/resolv.conf
nameserver 192.168.1.10
nameserver 192.168.1.11
```

Debian 12

→ https://192.168.1.11:9090/

Remove nginx running on port 80

```
vi /etc/nginx/sites-enabled/default # change 80 for 8080 for example so it doesn't occupy port 80 serving pihole
```

ref: https://www.reddit.com/r/pihole/comments/cf9efk/lighttpd_not_serving_up_web_interface/

install pihole

```
curl -sSL https://install.pi-hole.net | bash
pihole -a -p NEWPASS
reboot
```

```
timedatectl set-timezone Europe/Zurich
timedatectl

vi /etc/systemd/timesyncd.conf
```

```
[Time]
NTP=ntp.fortier-family.com
```

```
timedatectl set-ntp true
timedatectl status
systemctl restart systemd-timesyncd
```

```
vi /etc/pihole/custom.list
```

```
192.168.1.53 alpine.fortier-family.com
192.168.1.58 arch.fortier-family.com
192.168.1.80 cc.fortier-family.com
192.168.1.57 cleard.fortier-family.com
192.168.1.22 dc.fortier-family.com
192.168.1.65 debian.fortier-family.com
192.168.1.10 dns.fortier-family.com
192.168.1.11 dns2.fortier-family.com
192.168.1.61 endeavour.fortier-family.com
192.168.1.50 soc.fortier-family.com
192.168.1.70 unifi.fortier-family.com
192.168.1.20 proxmox.fortier-family.com
192.168.1.55 nixos.fortier-family.com
192.168.1.107 ds2413.fortier-family.com
192.168.1.105 ds409.fortier-family.com
192.168.1.30 ntp.fortier-family.com
192.168.1.68 rhel.fortier-family.com
192.168.1.42 kali.fortier-family.com
192.168.1.40 esxi01.fortier-family.com
192.168.1.69 frx.fortier-family.com
192.168.1.71 ntzghost.fortier-family.com
```

```
vi /etc/default/bb-wl18xx
USE_GENERATED_DNSMASQ=no
rm /var/lib/misc/dnsmasq.leases
touch /var/lib/misc/dnsmasq.leases
chown pihole:pihole /var/lib/misc/dnsmasq.leases
vi /etc/dnsmasq.d/SoftAp0
```

```
#cache-size=2048
#dhcp-leasefile=/var/run/dnsmasq.leases
reboot
```

Setup dhcp/fixed IP vi /etc/network/interfaces # The primary network interface auto eth0 iface eth0 inet static

```
address 192.168.1.11
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameserver 192.168.1.10 192.168.1.11
```

systemctl disable connman systemctl enable –now networking

Remove crap connman apt remove –purge connman reboot systemctl disable dnsmasq </code>

Update

```
 apt update && apt upgrade
cd /opt/scripts/tools/
git pull
./update_kernel.sh
reboot
```

https://elinux.org/Beagleboard:BeagleBoneBlack_Debian#i_take_full_responsibility_for_knowing_my_beagle_is_now_insecure

Pi-hole

```
curl -sSL https://install.pi-hole.net | bash
```

Unbound DNS

```
apt install unbound
```

- /etc/unbound/unbound.conf.d/unbound.conf

```
server:
    # If no logfile is specified, syslog is used
    # logfile: "/var/log/unbound/unbound.log"
    verbosity: 0

    interface: 127.0.0.1
    port: 5335
    do-ip4: yes
    do-udp: yes
    do-tcp: yes
```

```
    # May be set to yes if you have IPv6 connectivity
    do-ip6: no

    # You want to leave this to no unless you have *native* IPv6. With
6to4 and
    # Terredo tunnels your web browser should favor IPv4 for the same
reasons
    prefer-ip6: no

    # Use this only when you downloaded the list of primary root
servers!
    # If you use the default dns-root-data package, unbound will find
it automatically
    #root-hints: "/var/lib/unbound/root.hints"

    # Trust glue only if it is within the server's authority
    harden-glue: yes

    # Require DNSSEC data for trust-anchored zones, if such data is
absent, the zone becomes BOGUS
    harden-dnssec-stripped: yes

    # Don't use Capitalization randomization as it known to cause
DNSSEC issues sometimes
    # see
https://discourse.pi-hole.net/t/unbound-stubby-or-dnscrypt-proxy/9378
for further details
    use-caps-for-id: no

    # Reduce EDNS reassembly buffer size.
    # Suggested by the unbound man page to reduce fragmentation
reassembly problems
    edns-buffer-size: 1472

    # Perform prefetching of close to expired message cache entries
    # This only applies to domains that have been frequently queried
    prefetch: yes

    # One thread should be sufficient, can be increased on beefy
machines. In reality for most users running on small networks or on a
single machine, it should be unnecessary to seek performance
enhancement by increasing num-threads above 1.
    num-threads: 1

    # Ensure kernel buffer is large enough to not lose messages in
traffic spikes
    so-rcvbuf: 1m

    # Ensure privacy of local IP ranges
    private-address: 192.168.0.0/16
```

```
    private-address: 169.254.0.0/16
    private-address: 172.16.0.0/12
    private-address: 10.0.0.0/8
    private-address: fd00::/8
    private-address: fe80::/10
```

- /etc/unbound/unbound.conf.d/remote-control.conf

```
remote-control:
    control-enable: yes
    control-interface: /run/unbound.ctl
    control-use-cert: no
```

- /etc/unbound/unbound.conf.d/root-auto-trust-anchor-file.conf< code perl>

server:

```
  auto-trust-anchor-file: "/var/lib/unbound/root.key"
  trust-anchor-signaling: yes
```

</code> in http://192.168.1.11/admin Settings > DNS uncheck Google's and add **127.0.0.1#5335** as custom DNS 1

Troubleshoot DNS

```
systemctl stop cloud9.service
systemctl stop cloud9.socket
systemctl disable cloud9.service
systemctl disable cloud9.socket
```

Modify files like in https://services.haacksnetworking.org/2021/02/28/pihole-on-the-beagle-bone-black/
Also some references:
→ https://github.com/pi-hole/pi-hole/issues/1521
And:
→ https://discourse.pi-hole.net/t/new-install-dns-service-not-running/18644/11 And:
→ https://discourse.pi-hole.net/t/existing-dnsmasq-pi-hole/13533/6 In a nutshell:

```
vi /usr/bin/bb_dnsmasq_config.sh # comment cache-size line
vi /opt/scripts/boot/am335x_evm.sh # comment cache-size line too
vi /etc/default/bb-wl18xx # USE_GENERATED_DNSMASQ=no
systemctl disable dnsmask
apt remove dnsmasq
systemctl restart pihole-FTL

## in case of errors ...
systemctl disable wpa_supplicant
systemctl disable bonescript-autorun.service
systemctl stop pihole-FTL
rm /etc/dnsmasq.d/SoftAp0
touch /var/run/dnsmasq.leases
```

```
chown pihole /var/run/dnsmasq.leases
systemctl restart pihole-FTL
systemctl status pihole-FTL
```

PiAlert

Interface: http://192.168.1.11/pialert/
Reference: https://github.com/pucherot/Pi.Alert/blob/main/docs/INSTALL.md

```
 curl -sSL
https://github.com/pucherot/Pi.Alert/raw/main/install/pialert_install.sh |
bash
```

Unifi Controller (doesn't work)

→ https://www.ui.com/download/unifi/unifi-flex-hd

```
apt install apt-transport-https ca-certificates wget dirmngr gnupg gnupg2
software-properties-common multiarch-support
wget -qO - https://www.mongodb.org/static/pgp/server-3.4.asc |  apt-key add
-
echo "deb http://repo.mongodb.org/apt/debian jessie/mongodb-org/3.4 main" |
tee /etc/apt/sources.list.d/mongodb-org-3.4.list
wget http://security.debian.org/debian-
security/pool/updates/main/o/openssl/libssl1.0.0_1.0.1t-1+deb8u12_armhf.deb
dpkg -i libssl1.0.0_1.0.1t-1+deb8u12_armhf.deb
wget -qO - https://adoptopenjdk.jfrog.io/adoptopenjdk/api/gpg/key/public |
apt-key add -
add-apt-repository --yes https://adoptopenjdk.jfrog.io/adoptopenjdk/deb/
apt update
apt install adoptopenjdk-8-hotspot
echo "export JAVA_HOME=\"/usr/lib/jvm/adoptopenjdk-8-hotspot-amd64\""
>>/etc/profile
source /etc/profile
echo $JAVA_HOME
apt-key adv --keyserver keyserver.ubuntu.com --recv 06E85760C0A52C50
echo 'deb https://www.ui.com/downloads/unifi/debian stable ubiquiti' | tee
/etc/apt/sources.list.d/100-ubnt-unifi.list
apt update && apt install unifi
wget https://dl.ui.com/unifi/6.5.54/unifi_sysvinit_all.deb
apt install ./unifi_sysvinit_all.deb
```

Extra tools

```
apt install zsh htop ccze xrdp
```

Static IP

```
connmanctl services
*AO Wired                ethernet_1cba8ca24f0d_cable
connmanctl config ethernet_1cba8ca24f0d_cable --ipv4 manual 192.168.1.11
255.255.255.0 192.168.1.1 --nameservers 192.168.1.10
hostnamectl set-hostname dns2
```

Pi-Hole (doesn't work)

```
curl -sSL https://install.pi-hole.net | bash
```

if any issue:

```
pihole -r
```

Unbound DNS

```
apt install unbound
```

- /etc/unbound/unbound.conf.d/unbound.conf

```
server:
    # Basic Settings
    verbosity: 0
    interface: 127.0.0.1
    port: 5335
    do-ip4: yes
    do-udp: yes
    do-tcp: yes
    do-ip6: no
    prefer-ip6: no

    # Performance Tuning (optimized for BeagleBone Black)
    num-threads: 2
    msg-cache-size: 16m
    rrset-cache-size: 32m
    so-rcvbuf: 4m
    so-sndbuf: 4m
    outgoing-range: 2048
    num-queries-per-thread: 1024
    minimal-responses: yes
    prefetch: yes
    prefetch-key: yes
    serve-expired: yes
    serve-expired-ttl: 3600
    cache-min-ttl: 300
    cache-max-ttl: 86400
```

```
    # Security Settings
    harden-glue: yes
    harden-dnssec-stripped: yes
    harden-below-nxdomain: yes
    harden-referral-path: yes
    use-caps-for-id: no
    hide-identity: yes
    hide-version: yes
    qname-minimisation: yes

    # EDNS Settings
    edns-buffer-size: 1472

    # Root Server Configuration
    root-hints: "/var/lib/unbound/root.hints"

    # Private Networks
    private-address: 192.168.0.0/16
    private-address: 169.254.0.0/16
    private-address: 172.16.0.0/12
    private-address: 10.0.0.0/8
    private-address: fd00::/8
    private-address: fe80::/10

    # Aliases
    local-data: "srv0.fortier-family.com. IN CNAME kali2.fortier-
family.com."
```

- /etc/unbound/unbound.conf.d/remote-control.conf

```
remote-control:
    control-enable: yes
    control-interface: /run/unbound.ctl
    control-use-cert: no
```

- /etc/unbound/unbound.conf.d/root-auto-trust-anchor-file.conf

```
server:
    auto-trust-anchor-file: "/var/lib/unbound/root.key"
    trust-anchor-signaling: yes
```

```
curl -o /var/lib/unbound/root.hints
https://www.internic.net/domain/named.root
chown -R unbound:unbound /var/lib/unbound
unbound-checkconf
systemctl restart unbound
```

in http://192.168.1.11/ Settings > DNS add **127.0.0.1#5335** as custom DNS

Xrdp (not installed)

Config

```
systemctl enable --now xrdp
adduser xrdp ssl-cert
systemctl restart xrdp
```

Install Archlinux

Il vous faut une carte SSD et un lecteur sur une machine Linux déjà fonctionnelle

Préparation

```
dd if=/dev/zero of=/dev/mmcblk0 bs=1M count=8
```

partition the SD card:

```
fdisk /dev/mmcblk0
```

Type o. This will clear out any partitions on the drive.
Type n, then p for primary, 1 for the first partition on the drive, 2048 for the first sector, and then press ENTER to accept the default last sector.
Type w to write the partition table and exit
Format in ext4 filesystem:

```
mkfs.ext4 /dev/mmcblk0p1
```

Mount the card

```
cd /
mount /dev/mmcblk0p1 mnt
```

Copy to SD

```
wget http://os.archlinuxarm.org/os/ArchLinuxARM-am33x-latest.tar.gz
bsdtar -xpvf ArchLinuxARM-am33x-latest.tar.gz -C mnt && sync
```

U-boot

```
dd if=mnt/boot/MLO of=/dev/mmcblk1 count=1 seek=1 conv=notrunc bs=128k
```

```
dd if=mnt/boot/u-boot.img of=/dev/mmcblk0 count=2 seek=1 conv=notrunc
bs=384k
umount mnt
sync
```

First boot

Insert the card in the BBB then connect the network cable then, while holding the “user” button, insert the power plug. When all led are lighted up release the “user button”

Initialise paceman keys

```
ssh alarm@IP # pass: alarm
```

```
pacman-key --init
pacman-key --populate archlinuxarm
```

The BBB is working fully however on the SD card

Flash eMMC

Same steps as above but... with /dev/mmcblk1

```
dd if=/dev/zero of=/dev/mmcblk1 bs=1M count=8
fdisk /dev/mmcblk1
mkfs.ext4 /dev/mmcblk1p1
cd /
mount /dev/mmcblk1p1 mnt
wget http://os.archlinuxarm.org/os/ArchLinuxARM-am33x-latest.tar.gz
bsdtar -xpvf ArchLinuxARM-am33x-latest.tar.gz -C mnt && sync
dd if=mnt/boot/MLO of=/dev/mmcblk1 count=1 seek=1 conv=notrunc bs=128k
dd if=mnt/boot/u-boot.img of=/dev/mmcblk1 count=2 seek=1 conv=notrunc
bs=384k
umount mnt
sync
shutdown now
ssh alarm@IP # pass: alarm - root/root ( su )
pacman-key --init
pacman-key --populate archlinuxarm
```

Update&new toys

```
 pacman -Syu
```

```
pacman -S htop ccze dfc zsh vim base-devel git go #go for yay
```

Extra-config

hostname

```
hostnamectl set-hostname dns2
```

fixed IP

```
vi /etc/systemd/network/20-wired.network
```

```
[Match]
Name=eth0

[Network]
Address=192.168.1.11/24
Gateway=192.168.1.1
DNS=192.168.1.10
```

VIM über älles

```
 pacman -R vi
ln -s `which vim` /usr/bin/vi
```

AUR Helper

Let's install yay

```
su - alarm
mkdir /tmp/yay
curl https://aur.archlinux.org/cgit/aur.git/plain/PKGBUILD?h=yay >
/tmp/yay/PKGBUILD
cd /tmp/yay
makepkg
su
pacman -U yay*.xz
```

Pi-Hole

 as user **alarm** 

```
alarm@dns2 ~]$ yay -S pi-hole-server
:: Checking for conflicts...
:: Checking for inner conflicts...
[Repo:10]  libidn-1.38-1  bc-1.07.1-4  inetutils-2.2-1  logrotate-3.18.1-1
libmaxminddb-1.6.0-1  lmdb-0.9.29-1  python-3.9.9-1  python-ply-3.11-8
bind-9.16.23-1  lsof-4.94.0-1
[Repo Make:6]  hicolor-icon-theme-0.17-2  jsoncpp-1.9.4-1  libnsl-2.0.0-1
libuv-1.42.0-1  rhash-1.4.2-1  cmake-3.22.1-1
[Aur:2]  pi-hole-ftl-5.11-1  pi-hole-server-5.6-4

==> Remove make dependencies after install? [y/N]
  2 pi-hole-ftl                                (Build Files Exist)
  1 pi-hole-server                             (Build Files Exist)
==> Packages to cleanBuild?
==> [N]one [A]ll [Ab]ort [I]nstalled [No]tInstalled or (1 2 3, 1-3, ^4)
==> A
:: Deleting (1/2): /home/alarm/.cache/yay/pi-hole-ftl
:: Deleting (2/2): /home/alarm/.cache/yay/pi-hole-server
:: (1/2) Downloaded PKGBUILD: pi-hole-ftl
:: (2/2) Downloaded PKGBUILD: pi-hole-server
  2 pi-hole-ftl                                (Build Files Exist)
  1 pi-hole-server                             (Build Files Exist)
==> Diffs to show?
==> [N]one [A]ll [Ab]ort [I]nstalled [No]tInstalled or (1 2 3, 1-3, ^4)
==> N
```

coffee time 🤣
the compilation used to break @ 33%

```
[ 31%] Built target api
[ 32%] Building C object src/database/CMakeFiles/sqlite3.dir/shell.c.o
[ 33%] Building C object src/database/CMakeFiles/sqlite3.dir/sqlite3.c.o
/home/alarm/.cache/yay/pi-hole-ftl/src/FTL-5.11/src/database/sqlite3.c: In
function 'dbpageUpdate':
/home/alarm/.cache/yay/pi-hole-
ftl/src/FTL-5.11/src/database/sqlite3.c:206560:31: warning: comparison of
integer expressions of different signedness: 'Pgno' {aka 'unsigned int'} and
'int' [-Wsign-compare]
206560 |   if( pgno<1 || pBt==0 || pgno>(int)sqlite3BtreeLastPage(pBt) ){
       |                               ^
{standard input}: Assembler messages:
{standard input}: Error: open CFI at the end of file; missing .cfi_endproc
directive
...
```

Looks like the issue is the lack of memory to using
https://docs.rackspace.com/support/how-to/create-a-linux-swap-file/ I added one GB of swap on /dev/mmcblk0p1
Once installed, start/enable pihole-FTL service

```
systemctl start pihole-FTL
```

It will fail silently thanks to SystemD and its systemd-resolved.service…

```
 vi /etc/systemd/resolved.conf
```

```
[Resolve]
DNSStubListener=no
```

Restart both…

```
 systemctl restart systemd-resolved pihole-FTL
```

Php

- Install

  ```
   yay -S php-sqlite
  ```

- /etc/php/php.ini

  ```
  [...]
  extension=pdo_sqlite
  [...]
  extension=sockets
  [...]
  extension=sqlite3
  [...]
  ```

Lighttpd

```
 yay -S lighttpd php-cgi
cp /usr/share/pihole/configs/lighttpd.example.conf
/etc/lighttpd/lighttpd.conf
systemctl enable --now lighttpd
```

Hosts

```
 vi /etc/hosts
```

```
127.0.0.1               localhost
192.168.1.11   pi.hole dns2
```

Unbound

Let's install a real recursive DNS

Install

```
yay -S unbound
```

Config

→ https://docs.pi-hole.net/guides/dns/unbound/
In /etc/unbound/unbound.conf.d/pi-hole.conf

```
server:
    # If no logfile is specified, syslog is used
    # logfile: "/var/log/unbound/unbound.log"
    verbosity: 0

    interface: 127.0.0.1
    port: 5335
    do-ip4: yes
    do-udp: yes
    do-tcp: yes

    # May be set to yes if you have IPv6 connectivity
    do-ip6: no

    # You want to leave this to no unless you have *native* IPv6. With 6to4
and
    # Terredo tunnels your web browser should favor IPv4 for the same
reasons
    prefer-ip6: no

    # Use this only when you downloaded the list of primary root servers!
    # If you use the default dns-root-data package, unbound will find it
automatically
    #root-hints: "/var/lib/unbound/root.hints"

    # Trust glue only if it is within the server's authority
    harden-glue: yes

    # Require DNSSEC data for trust-anchored zones, if such data is absent,
the zone becomes BOGUS
    harden-dnssec-stripped: yes

    # Don't use Capitalization randomization as it known to cause DNSSEC
issues sometimes
    # see
https://discourse.pi-hole.net/t/unbound-stubby-or-dnscrypt-proxy/9378 for
further details
```

```
    use-caps-for-id: no

    # Reduce EDNS reassembly buffer size.
    # Suggested by the unbound man page to reduce fragmentation reassembly
problems
    edns-buffer-size: 1472

    # Perform prefetching of close to expired message cache entries
    # This only applies to domains that have been frequently queried
    prefetch: yes

    # One thread should be sufficient, can be increased on beefy machines.
In reality for most users running on small networks or on a single machine,
it should be unnecessary to seek performance enhancement by increasing num-
threads above 1.
    num-threads: 1

    # Ensure kernel buffer is large enough to not lose messages in traffic
spikes
    so-rcvbuf: 1m

    # Ensure privacy of local IP ranges
    private-address: 192.168.0.0/16
    private-address: 169.254.0.0/16
    private-address: 172.16.0.0/12
    private-address: 10.0.0.0/8
    private-address: fd00::/8
    private-address: fe80::/10
```

```
systemctl enable unbound
```

Add your own entries

Whether through the weeb interface or... through

```
vi /etc/pihole/custom.list
```

```
192.168.1.53 alpine.fortier-family.com
192.168.1.58 arch.fortier-family.com
192.168.1.80 cc.fortier-family.com
192.168.1.57 cleard.fortier-family.com
192.168.1.22 dc.fortier-family.com
192.168.1.65 debian.fortier-family.com
192.168.1.10 dns.fortier-family.com
192.168.1.11 dns2.fortier-family.com
192.168.1.61 endeavour.fortier-family.com
192.168.1.50 soc.fortier-family.com
192.168.1.70 unifi.fortier-family.com
192.168.1.20 proxmox.fortier-family.com
```

```
192.168.1.55 nixos.fortier-family.com
192.168.1.107 ds2413.fortier-family.com
192.168.1.105 ds409.fortier-family.com
192.168.1.30 ntp.fortier-family.com
192.168.1.68 rhel.fortier-family.com
192.168.1.42 kali.fortier-family.com
192.168.1.40 esxi01.fortier-family.com
192.168.1.69 frx.fortier-family.com
```

Now just enter 127.0.0.1#5335 in Settings > DNS (upstream DNS) Custom 1, unchecking any upstream DNS previously setup.

References

- https://www.how2shout.com/linux/commands-to-install-xrdp-server-on-debian-11-bullseye-linux/
- http://derekmolloy.ie/set-ip-address-to-be-static-on-the-beaglebone-black/
- https://forum.digikey.com/t/static-ip-on-beaglebone-black/4497
- https://devopstales.github.io/linux/install-unifi-controller/ [not working on Debian's image...]
- https://discourse.pi-hole.net/t/pihole-dns-service-not-running-and-ftl-offline-after-v5-installation/31262/10
- https://www.youtube.com/watch?v=FnFtWsZ8IP0
- https://frdmtoplay.com/running-pi-hole-on-the-beagle-bone-black/
- https://discourse.pi-hole.net/t/update-from-v5-to-v5-2-but-pihole-status-shows-dns-service-is-not-listening/44316
- https://archlinuxarm.org/platforms/armv7/ti/beaglebone-black
- https://wiki.archlinux.org/title/Pi-hole
- https://docs.rackspace.com/support/how-to/create-a-linux-swap-file/