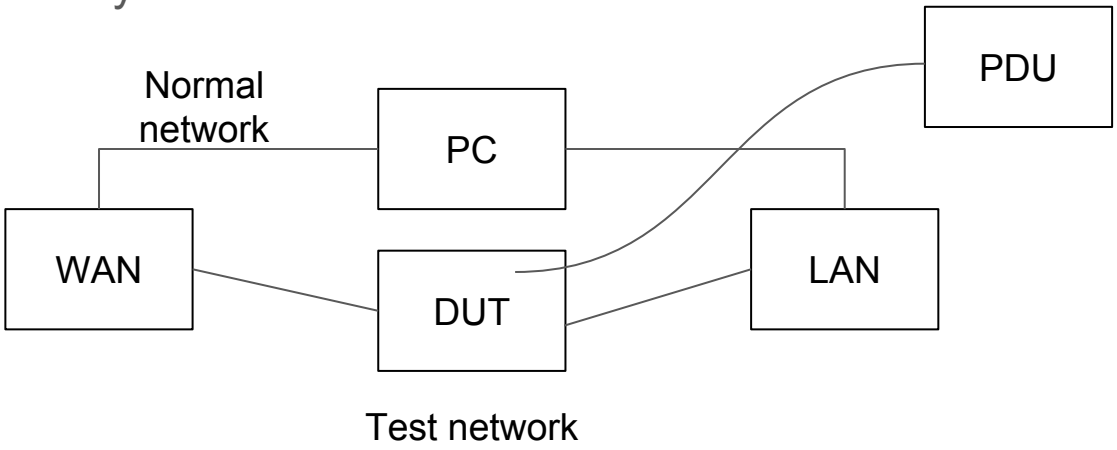
Boardfarm test tools

Matthew McClintock <matthew@mcclintock.net>
Eric Schultz <eschultz@prplfoundation.org>

https://github.com/qca/boardfarm

High level overview of boardfarm scripts

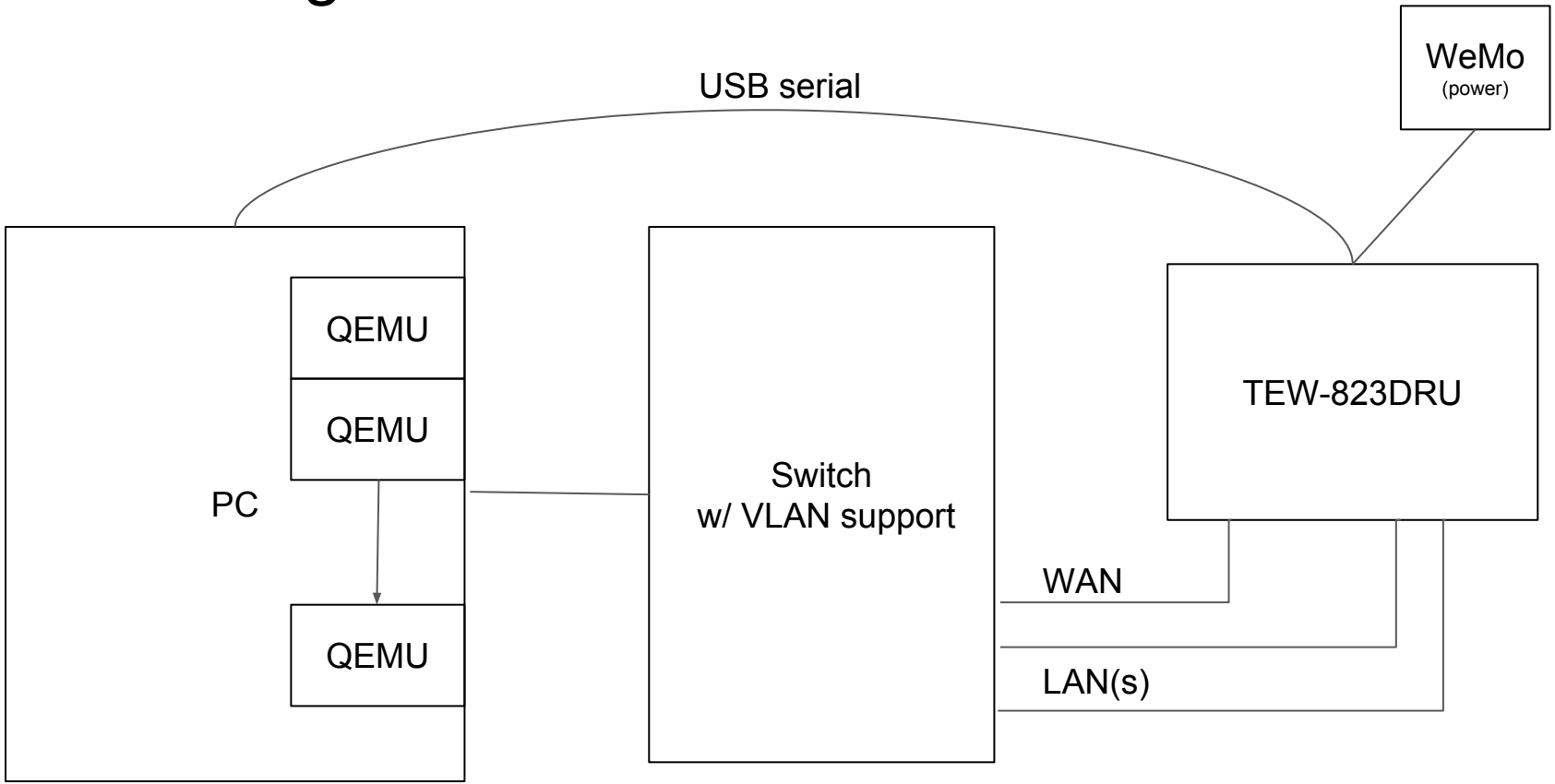
- Python based tool, heavily relies on pexpect for tests
- One machine drives the test (runs the python script from the repo)
- One machine for the WAN, and additional machines for LAN/WLAN/etc
- Connection to the device under test (typically serial)
- Method to power cycle the board is ideal



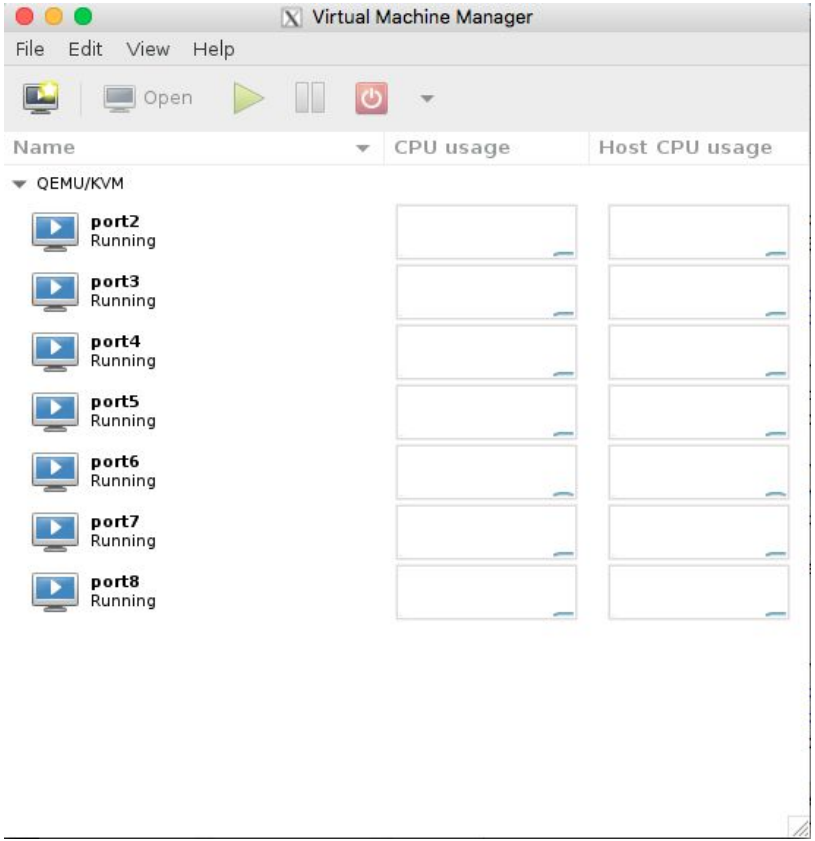
Before…

And After...

Overall diagram



Qemu images for devices connected to boards



Boardfarm config

```
[mattsm@pandora:~/git/boardfarm-config] master(+37/-15) ± cat boardfarm_config.json | grep -v password
{
    "tew-823dru-1": {
        "board_type": "tew-823dru",
        "conn_cmd": "cu -l /dev/serial/by-path/pci-0000:00:1a.0-usb-0:1.4.1:1.0-port0 -s 115200",
        "location": "austin",
        "powerip": "192.168.100.10",
        "power_username": "admin",
        "powerport":"1",
        "devices": [
                {
                        "type": "debian",
                        "name": "wan",
                        "ipaddr": "port2",
                        "color": "cyan"
                },
                {
                        "type": "debian",
                        "name": "lan",
                        "ipaddr": "port3",
                        "color": "blue",
                        "options": "tftpd-server"
                },
                {
                        "type": "debian",
                        "name": "lan2",
                        "ipaddr": "port4",
                        "color": "blue"
```

Bill of materials

- If you have a computer with a free ethernet port and a device to test and it's capable of running at least 1 ideally 2+ qemu guests you can build a test station on the cheap

PDU (WeMo)	~ $40
2 USB Ethernet or a Switch with VLANs	~ $30
USB serial	~ $6

- Scales well, a 16 port switch is only ~$60 which can drive 7 machines with a WAN and a LAN connection only with one uplink and one spare port for a real network driven PDU

Adding new board support

2 devices/board_decider.py — View

```
@@ -19,7 +19,7 @@ def board(model, **kwargs):
     different memory addresses, and must be handled differently.
     '''
     if model in ("db120", "ap135", "ap143", "ap147", "ap152", "ap151",
-                 "ap151-16M", "ap143", "ap152-8M"):
+                 "ap151-16M", "ap143", "ap152-8M", "tew-823dru"):
         return qcom_mips.QcomMipsRouter(model, **kwargs)

     if model in ("ipq8066", "db149", "ap145", "ap148", "ap148-osprey",
```

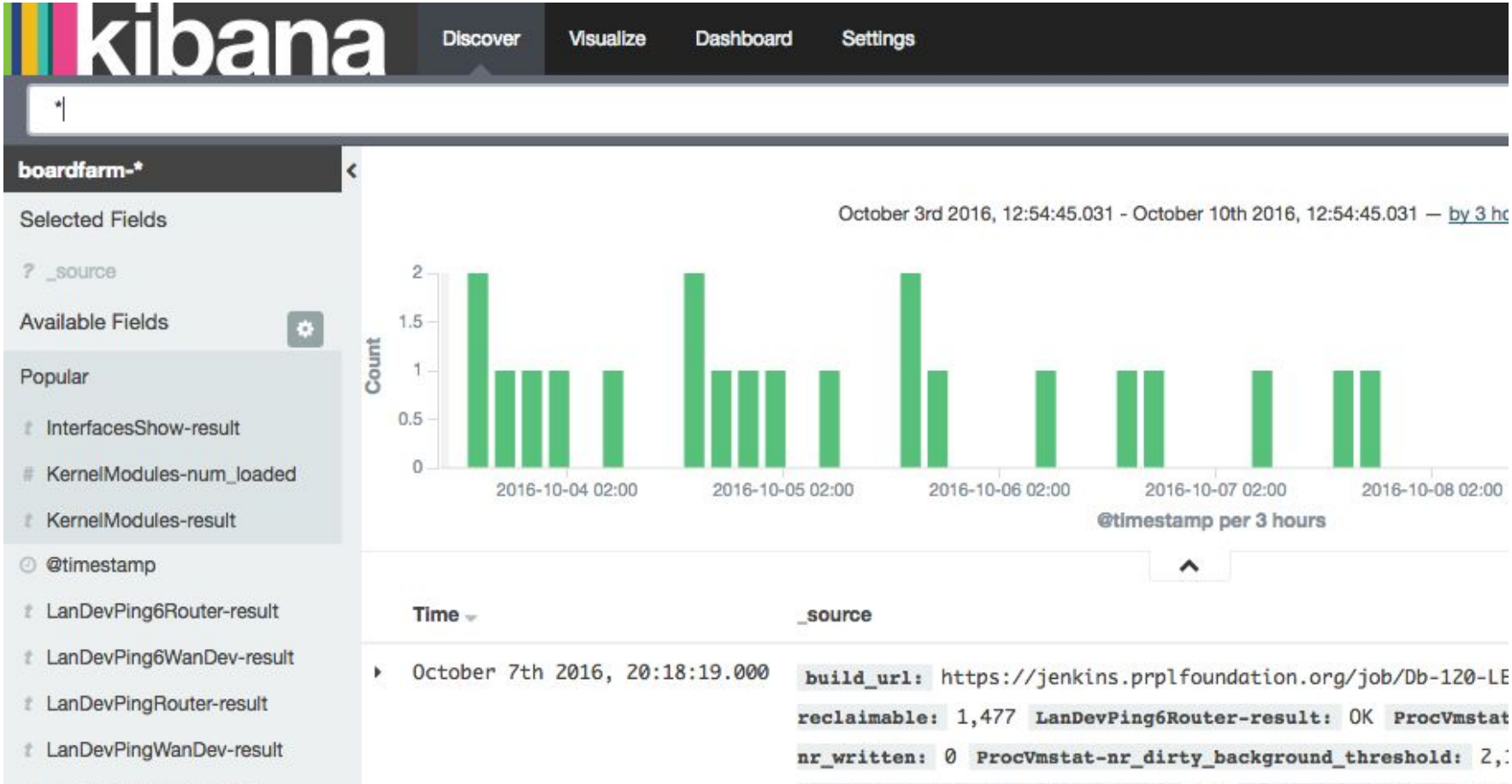
4 devices/qcom_mips.py — View

```
@@ -32,6 +32,10 @@ def check_memory_addresses(self):
         elif self.model in ("db120", "ap143", "ap151", "ap152-8M"):
             self.kernel_addr = "0x9f680000"
             self.rootfs_addr = "0x9f050000"
+        elif self.model in ("tew-823dru"):
+            self.kernel_addr = "0x9f040000"
+            # rootfs undefined so we throw exception when trying to
+            # write for the time being
```

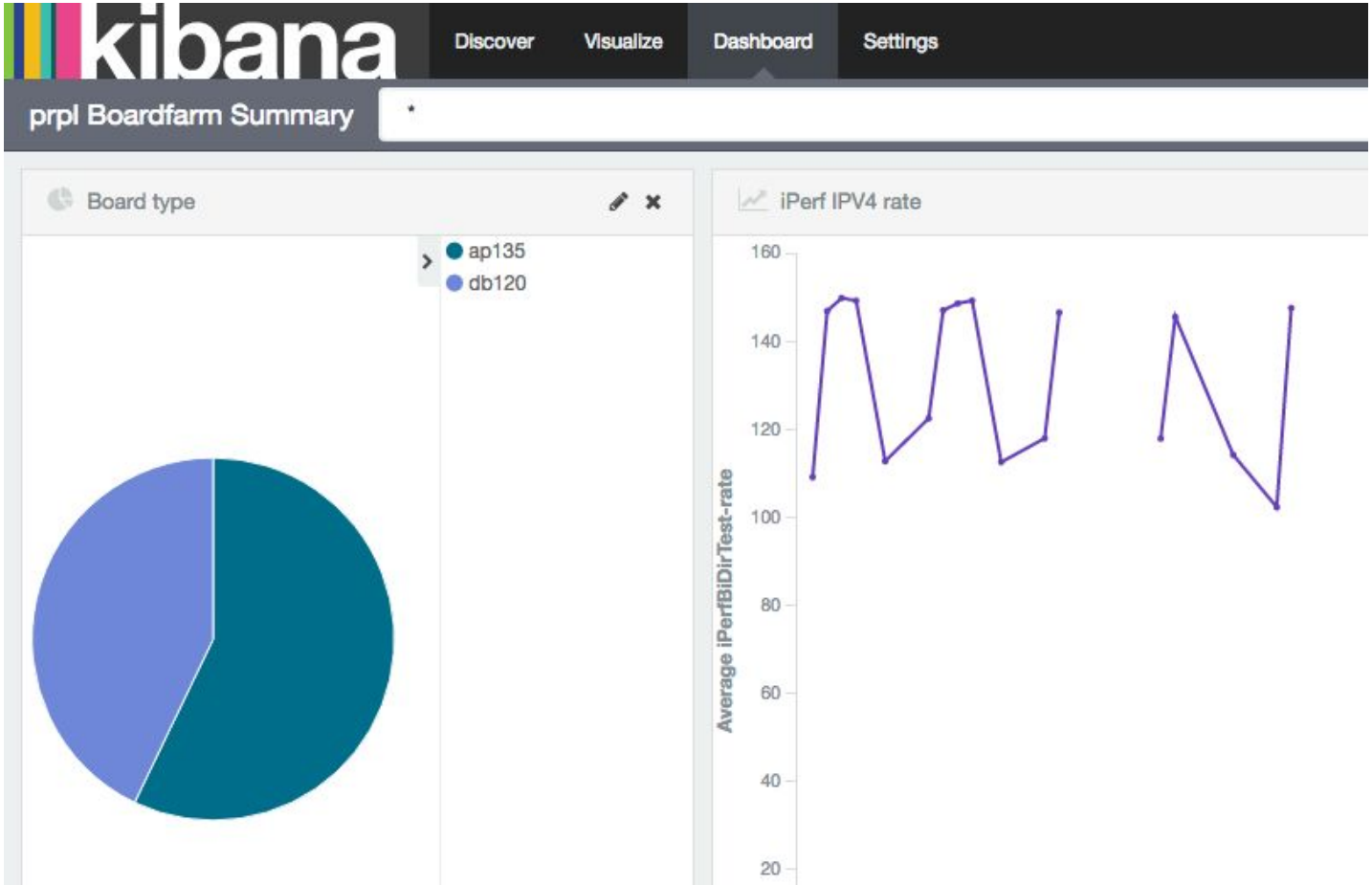
Example test

```
[mattsm@pandora:~/git/boardfarm] master(+1/-0,3)* ± cat tests/iperf_all_lan_ports.py | grep -v '#' | grep -v ^$
import rootfs_boot
import pexpect
import time
from devices import board, wan, lan, wlan, prompt
class iPerfAllLanPorts(rootfs_boot.RootFSBootTest):
    '''Runs iPerf on all LAN ports -> WAN ports'''
    def runTest(self):
        wan.sendline('iperf -s')
        devices = [ getattr(self.config, d) for d in self.config.devices if d.startswith('lan') ]
        for d in devices:
            d.sendline('dhclient -r eth1')
            d.expect(prompt)
            d.sendline('dhclient eth1')
            d.expect(prompt)
        for d in devices:
            d.sendline('iperf -c 192.168.0.1')
            i = d.expect(['Client connecting to 192.168.0.1', pexpect.TIMEOUT], timeout=5)
            if i == 2:
                print("Failed to connect to server on %s" % d.name)
        time.sleep(60)
        for d in devices:
            i = d.expect([pexpect.TIMEOUT] + prompt)
            if i == 0:
                print("iperf failed to finish on %s" % d.name)
        wan.sendcontrol('c')
        wan.expect(prompt)
```

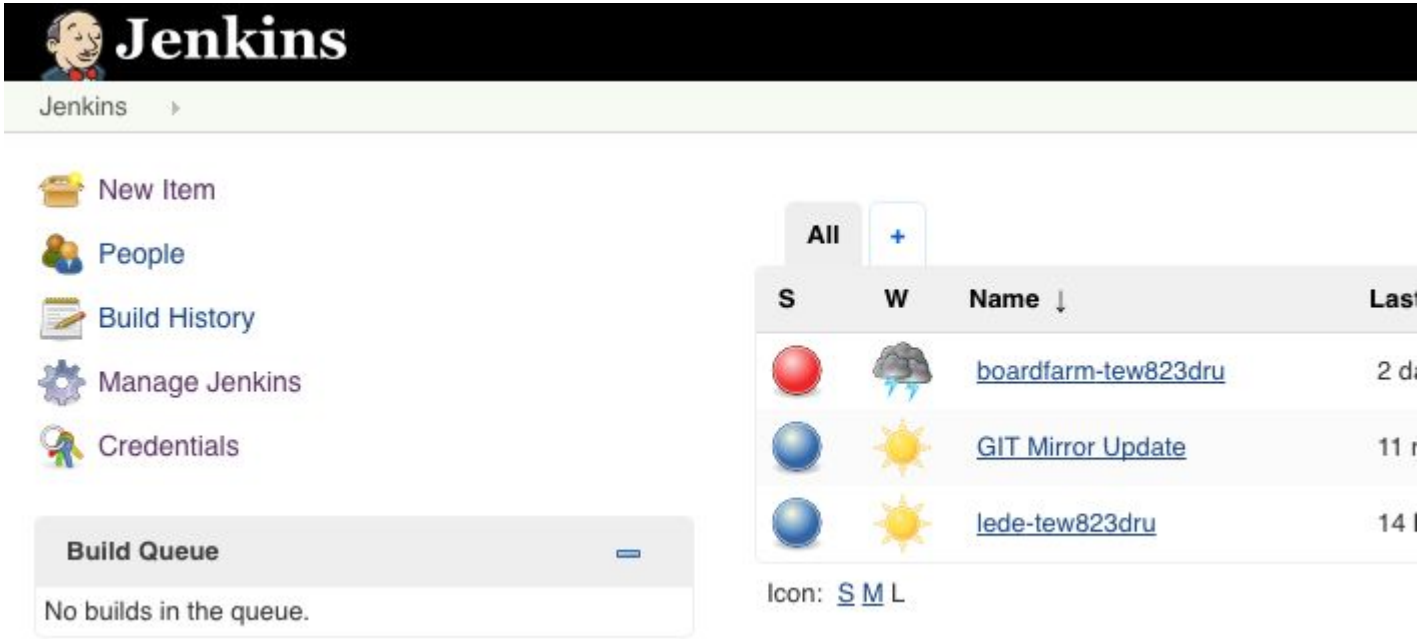
Kibana


kibana
Discover
Visualize
Dashboard
Settings
*
boardfarm-*
Selected Fields
? _source
Available Fields
Popular
t InterfacesShow-result
KernelModules-num_loaded
t KernelModules-result
@timestamp
t LanDevPing6Router-result
t LanDevPing6WanDev-result
t LanDevPingRouter-result
t LanDevPingWanDev-result
October 3rd 2016, 12:54:45.031 - October 10th 2016, 12:54:45.031 — by 3 hc
Count
2
1.5
1
0.5
0
2016-10-04 02:00
2016-10-05 02:00
2016-10-06 02:00
2016-10-07 02:00
2016-10-08 02:00
@timestamp per 3 hours
Time
_source
October 7th 2016, 20:18:19.000
build_url: https://jenkins.prplfoundation.org/job/Db-120-LE
reclaimable: 1,477 LanDevPing6Router-result: OK ProcVmstat
nr_written: 0 ProcVmstat-nr_dirty_background_threshold: 2,

Kibana dashboard



Next... automated tests for pull requests



Next... automated tests for pull requests

Execute shell

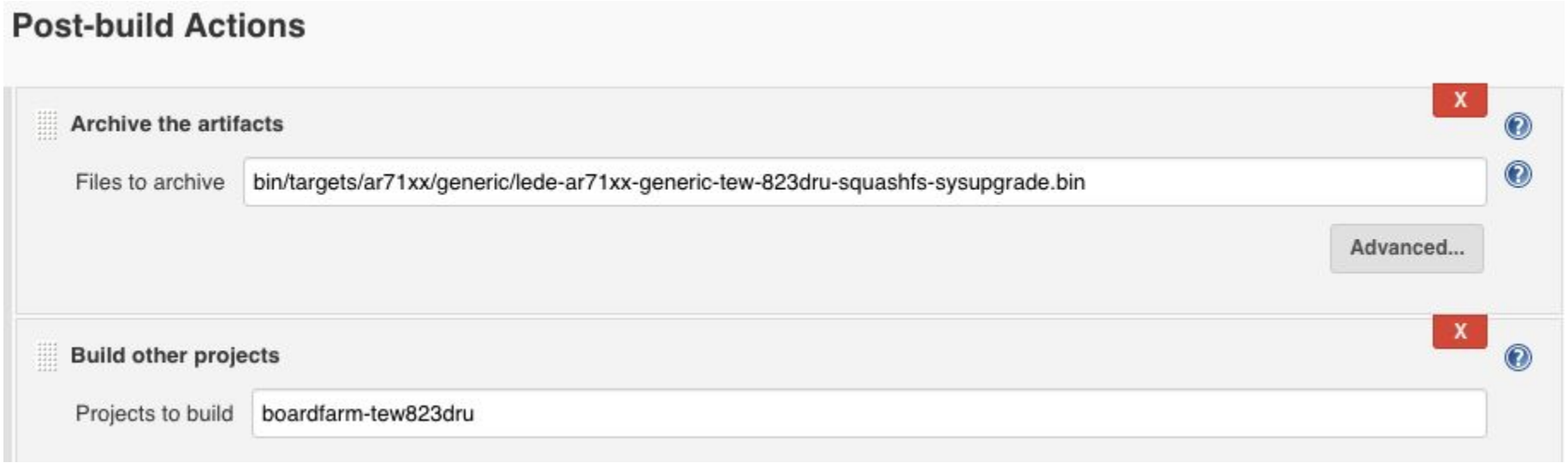
Command

```
cat << EOF >> .config
CONFIG_TARGET_ar71xx=y
CONFIG_TARGET_ar71xx_generic=y
CONFIG_TARGET_ar71xx_generic_DEVICE_TEW823DRU=y
CONFIG_PACKAGE_rpcd=y
CONFIG_PACKAGE_rpcd-mod-file=y
CONFIG_PACKAGE_rpcd-mod-iwinfo=y
CONFIG_PACKAGE_rpcd-mod-rpcsys=y
CONFIG_PACKAGE_uhttpd=y
CONFIG_PACKAGE_uhttpd-mod-ubus=y
EOF

make defconfig

make -j24
```

Next... automated tests for pull requests

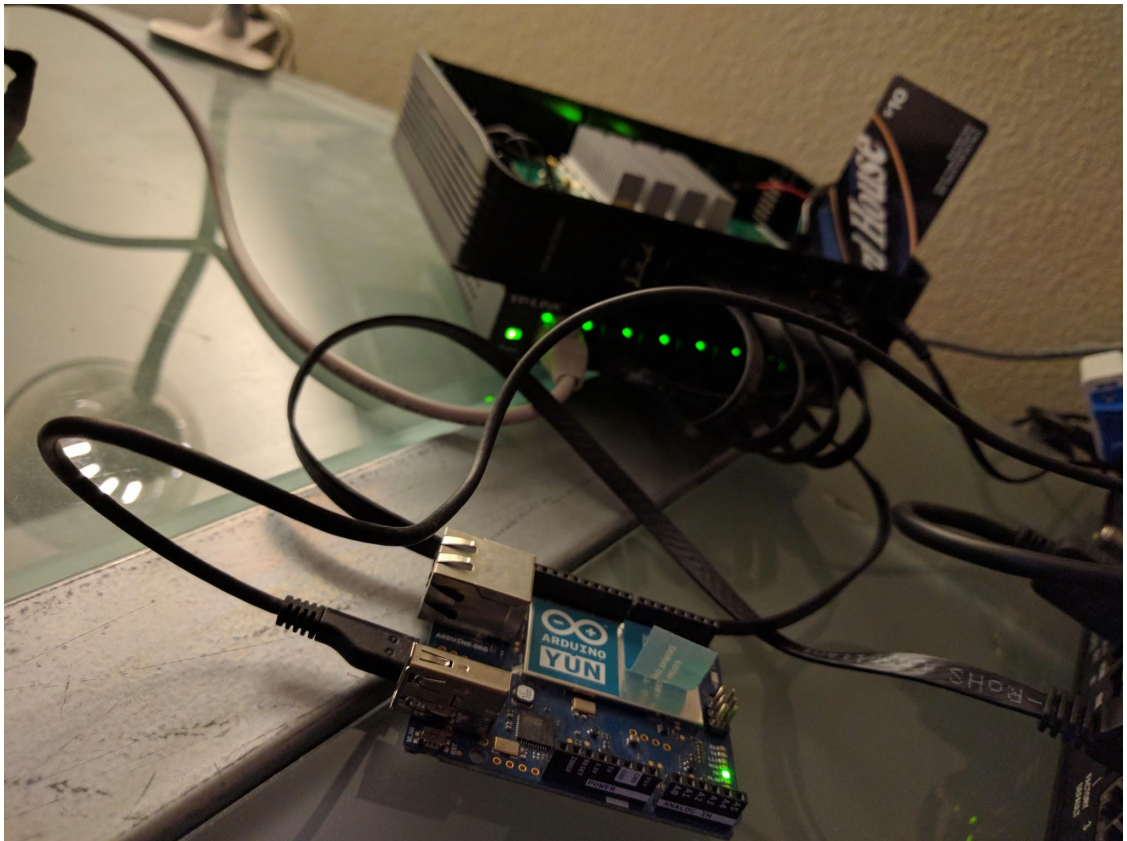


Next... automated tests for pull requests

Next... automated tests for pull requests

- Reporting data
  - Kibana
    - Separate servers or one common server?
  - Email
    - New lists or small other lists?
  - Bug reporting system
    - Script included in repo to upload from command line for automated purposes and format standardization?

Next… wireless testing



prpl, prplwrt and Boardfarm

- prpl
  - A non-profit member-driven community open source organization
  - One PEG (working group) is focused on OpenWrt/LEDE
- prplwrt
  - prpl Engineering Group (PEG) focusing on collaboration between members of prpl with each other and OpenWrt/LEDE community
  - One topic prplwrt focuses on is improving the Boardfarm project and running an instance of Boardfarm which we'll talk more about later

prplwrt Boardfarm Instance

- A public solution to a problem: no automated testing of OpenWrt/LEDE changes on real hardware
- Run daily tests of OpenWrt CC, DD and LEDE on a set of hardware
  - Currently three devices, with three more on their way
  - Donations are welcome and encouraged! (contact eschultz@prplfoundation.org)
- Use Raspberry Pi's (for now)
- Jenkins SSH's into private network, runs boardfarm tests and receives results (view at jenkins.prplfoundation.org)
- Some scripts for recreating is at prpl Foundation Github

How to get involved?

- http://github.com/qca/boardfarm - use it, add support for a device
- Donate a device to the prpl Boardfarm Instance (email eschultz@prplfoundation.org)
- Boardfarm discussion list: boardfarm@lists.prplfoundation.org
- #boardfarm on Freenode
- prplwrt Weekly Meetings to discuss Boardfarm:
  - 9AM PT, every Thursday
  - Boardfarm is a standing topic
  - Instructions for joining are sent to the list (Can join via WebRTC supported browser or via phone)