Teaching Your Toaster New Tricks

Or doing cool things with IoT

About Me

- About me
  - Student Researcher at Cal Poly Pomona– Learn by doing!
  - Focus on Internet of Things and Embedded Devices
  - Participate in CCDC, CPTC, and CTF competitions regularly
  - 3 years of active research in embedded devices

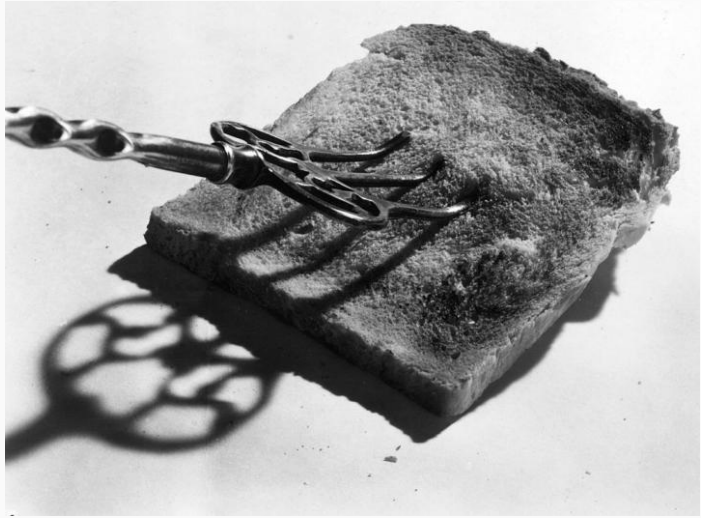
Agenda

- Look at the various types of devices that are available
- Find ways to make use of End of Life devices
- Find better ways to make “smart” devices

- Profit? Or end up with a IoToaster ***//***

Lets clear things up

https://www.technologyreview.com/s/400889/internet-on-a-chip/




HDTV
VIP722 DVR

Lets clear things up

Then there was….

Lets clear things up

And the future holds….

But this is all you get

The Victims...

- Routers
- Cameras
- NASes
- Travel Routers/Hotspots
- (WeMo) Coffee Maker
- Door Locks
- (WeMo/D-Link/TP-Link) Power Outlets
- (WeMo) Air Purifier / Cooler
- Drones (Parrot, Elfie, Generic)
- “Smart” TVs

Attack of the Clones

- Many IoT devices are based on reference models or are clones
- Cheaper to develop and release but doesn’t mean more secure

Dividing Everything Up

"Customizable Firmware"

- Asus **N16**, **N66**, and **AC88**
- GL.iNet **AR150** and **300N, AR300**
- WeMo ***Outlet***, ***Crockpot***, ***Coffee Maker***, *and* ***Air***
- TP Link **TL-WR710N** and **TL-WDR3600, HS100**
- HooToo **TM-02**
- Netgear **AC3200**
- Fosscam **Wifi Camera Clones**

"R/W Systems"

- *Parrot Drones*
- *WD* ***My Cloud*** *(Pure Debian!)*
- *QNAP* ***TS-251***

Why Divide Up Devices?

- Ensure we know what we're dealing with and what we will have to repair
- Level of Effort
- Identify what will be required to access the device
- Identify possible security issues as entry points

Parrot Drones

- Variety of drones available
- Relatively cheap
- Consistent Specs Advertized:
  - 1GB of RAM
  - 1ghz “Dual Core” Processor
- Actually:
  - 256-512MB of RAM and 400mhz Processor
- Great Marketing!

http://www.cpp.edu/~polysec/UAV/

Expectations

Expectations

Reality

Normal Use

- Phone App connects via WiFi
- Transfer data from the drone via FTP and AR-Stream Protocol
- ~~Emergency~~ Attack Mode?!

Gaining Access

```
bash-4.3$ telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.



BusyBox v1.14.0 () built-in shell (ash)
Enter 'help' for a list of built-in commands.

# ls
README          dev             firmware        mnt             sbin            usr
```

Why is this still a thing?

- So much is *“right”* with Parrot Drone Systems
- As other talks have shown – it runs telnet and ftp and random other ports – as we see “bash proxy”.
- Factory reset doesn’t factory reset anything except config.ini.
- **Firmware modification should not be made 60ft in the air!**

```
# IP_ADDR - this target IP address using CIDR notation:
# <target-ip>/<target-bitmask>
#
# For example:
echo init started...
```

Good use of Examples

```
## Misc Linux behaviour
# Flush data to disk more often
sysctl -w vm.dirty_expire_centisecs=200
sysctl -w vm.dirty_writeback_centisecs=200

# Initialize a list of arguments to pass to program.elf
echo "" > /tmp/.program.elf.arguments
echo "-360p.slices 0" >> /tmp/.program.elf.arguments
echo "-live.tcp"      >> /tmp/.program.elf.arguments

#/bin/ltt.sh
#/bin/start_trace.sh 1
/bin/wifi_setup.sh&
/sbin/udevd.sh
source /bin/parallel-stream.sh
#disabling kernel messages
echo 0 > /proc/sysrq-trigger
/bin/factory_reset_cb&
/bin/memory_check.sh &
# Start a proxy which executes Bash commands from program.elf
mkfifo /tmp/.bashproxyfifo.in
mkfifo /tmp/.bashproxyfifo.out
/bin/bashproxy /tmp/.bashproxyfifo.in /tmp/.bashproxyfifo.out &
# Copying licenses file to ftp directory so it can be seen by users.
cp /licenses/* /data/video
export I2C_DEVICE=/dev/i2c-1
i2c_cmd 0x48 0x19 0x02
/bin/init_gpios.sh
/bin/check_update.sh

# Force USB host mode - Set host_req bit (D1) and Session bit
devmem2 0x480ab060 w 0x7070003

# BEGIN PATCH
/bin/overrider.sh
mount /dev/sda1 /mnt 2>/dev/null
/mnt/cnf/init.sh
export PATH=/opt/sbin:/opt/bin:/sbin:/usr/sbin:/bin:/usr/bin
source /root/.profile
# END PATCH
```

init scripts? Nah

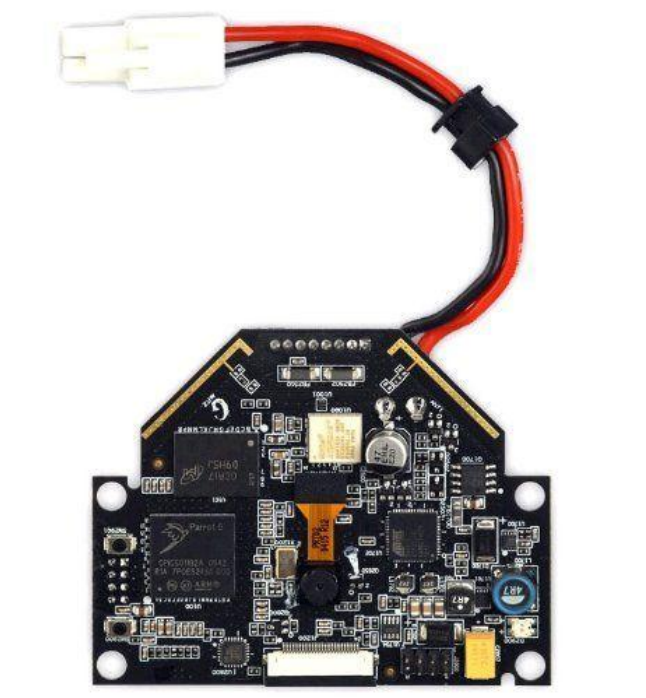
Glad we have debugging available!

This seems safe

By the power of greyskull!

What does that mean?

- Easy modification and exploitation of drones
- Perform modification on any local Parrot drones
- Communicate between Drones (multiplayer)
  - Stop drones
  - File Transfer / Take-Over
- Malware Upload / Credential Theft



killall program.elf?

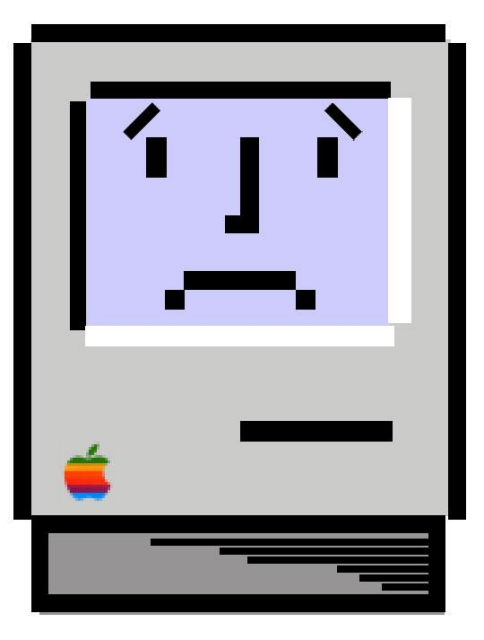
- Drone runs out of program.elf
- Everything else is just linux.
- Pretty sure this is what they mean by fully upgradable

- If you upgrade the firmware or just stop program.elf….

Improvements?

- Use OpenWRT
  - Compiled…
- BuildRoot
  - Compiled
  - Upload Directories

And…

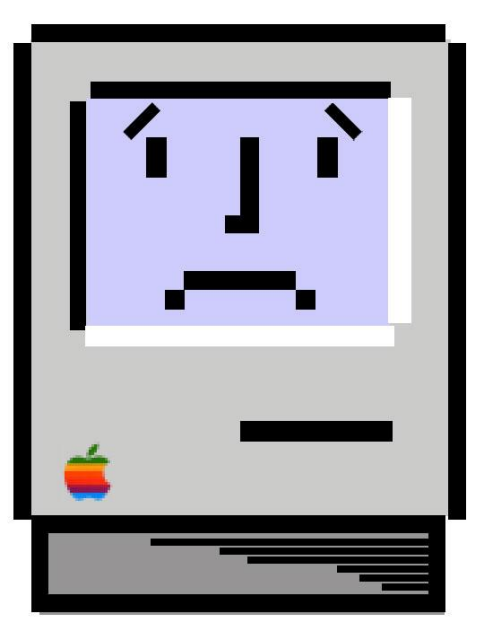


What went wrong?

- Build was set up after specific kernel / ulibc configurations
- No easy way to replace the system without taking up too much space
- Possibility of the brick

Try again!

- Compile Statically?



“optware”

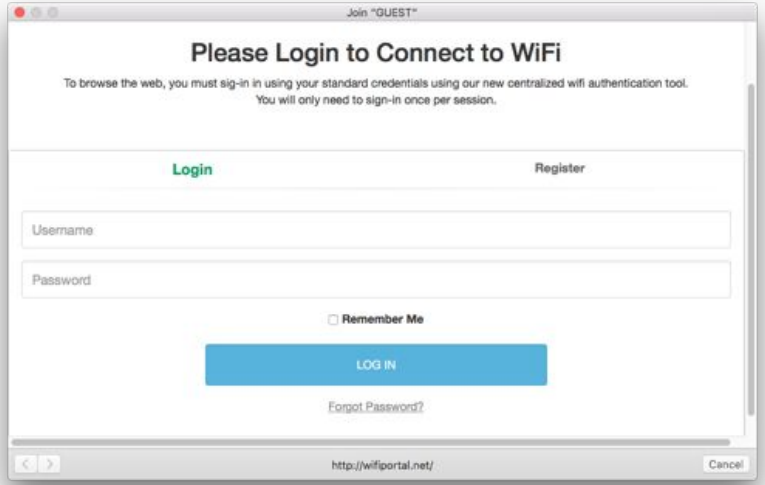
- All components patched to run out of /opt/
- Next Generation is: Entware-NG

- Plenty of packages, works everywhere


Stand back
There's science in this shit.

Ideas!

- Why couldn't we return this? With "improved" firmware?
- Download files to people's phones or tables.
- Mobile Captive Portal
- Drive by Drone Capture and Pivot



Captive Portals: Things Learned

- Most operating systems now have built in handling of captive portals.
- On latest platforms this interface is restricted

- However, on Windows and iOS you can have links that will allow people to open up an unrestricted browser
- Time to send some files!

Drone ⟷ Drone

- Parrot Drones have a unused featured called “Multi-Player”
- Allows drones to connect to a shared network or each other easily
- This also allows us to connect to drones and take them over
    - Drones are configured with IPTables but only flight control is blocked
    - Telnet and ftp are enabled and not blocked, allowing us to transfer and run payloads

```
$ telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.



BusyBox v1.14.0 () built-in shell (ash)
Enter 'help' for a list of built-in commands.

# ps | grep -v "\["
  PID USER       VSZ STAT COMMAND
    1 root      2736 S    init
  593 root      1788 S <  udevd --daemon
  598 root      1788 S <  udevd --daemon
  601 root      1788 S <  udevd --daemon
  795 root      1672 S    /bin/factory_reset_cb
  796 root      2744 S    /bin/sh /bin/memory_check.sh
  800 root      1676 S    /bin/bashproxy /tmp/.bashproxyfifo.in /tmp/.bashproxyfifo.out
  832 root      2736 S    inetd
  833 root      2736 S    /bin/sh /bin/program.elf.respawner.sh -360p.slices 0 -live.tcp
  836 root     84780 S    /bin/program.elf -360p.slices
  837 root      2736 S    init
  838 root      2736 S    init
  839 root      2736 S    /sbin/syslogd -n -m 0
  840 root      2736 S    /sbin/klogd -n
  951 root      2832 S    telnetd -l /bin/sh
  953 root      2740 S    udhcpd /tmp/udhcpd.conf
  963 root      1540 S    /bin/parrotauthdaemon
 1108 root      2740 S    /bin/sh
 1122 root      2740 S    /bin/sh
 1144 root      2604 S    sleep 10
 1145 root      2740 S    /bin/sh
 1146 root      2824 R    ps
# uname -a
Linux uclibc 2.6.32.9-g980dab2 #1 PREEMPT Mon Sep 16 11:50:23 CEST 2013 armv7l GNU/Linux
# lsmod
Module                  Size  Used by    Tainted: G
ov7670                  9943  1
soc1040                20296  2
omap3_isp              95903  9
ar6000                278079  0
#
```

WD MyCloud

- “With its robust software…
- Its Just Debian!
- Really..
  “Firmware Updates” are .deb packages!


BUT WAIT
Baking Soda
THERE'S MORE

Root?

- We don’t even have to try
- Web UI is fully optimized PHP (still)
- Multiple vulnerabilities in the Web UI.
    - Old: Status Checker run arbitrary Commands
    - http://wdmycloud.local/api/1.0/rest/safepoint_getstatus?handle=“$(telnetd)”
- New: Firmware Updater still allows command injection

Fun with Debian

- Restore the Debian repos, you have a fully functional arm Debian box.
- Upgrade or install anything you would like!
- Want to use Kali Tools? Sure thing!

No such thing as factory!

One thing we've seen so far with all these R/W devices.

- Factory Reset is just a name. IT DOES NOTHING… EVER...
- WD MyCloud factory reset does not restore Web UI files, does not reset most content on the drive.
- You want persistence... This is how you get persistence.

How did we find out?

Great News for Us!

- Remove WD’s features
- Low-Powered Server
- Network Monitor?

Possibilities are almost endless with one caveat - the kernel has been customized

Great News for Us!

- Remove WD’s features
- Low-Powered Server
- Network Monitor?

Possibilities are almost endless with one caveat - the kernel has been customized

240 days continuous uptime running bro via a tap

The other option…

- DD-WRT, OpenWRT, LEDE
- Firmware compresses extremely well
- (Usually) Easily unbricked, easily updated, easy maintenance
- Deploy to one system or dozens of all types, sizes, and kinds

Good and Bad

- The good: You can setup packages, resources to always run, and restore on failure.
- The bad: You are stuck with a set of packages and resources.
- The really bad: Not all devices are the same – even if they have the same chip! Fixes often required to setup a device (but upgrades are easier)

RA RT5350(F)

Why?

- Used by WeMo and dozens of other IoT platforms
- Usually has accessible UART (Serial)

Specs:

- 16MB flash, 32MB ram
- ~360mhz processor
- 802.11n 2.4ghz
- 4 port 10/100 switch (support)
- 1 usb
- GPIO

Plenty of Open Devices

- VoCore 1

  – Runs OpenWRT from the start, no need to provide additional patches

- HooToo Devices (TM-02)

  – Fully supported by OpenWRT, simply needs a initial “factory image”

Back to this...

```
$ nmap 10.10.10.254 -p 22,23; telnet 10.10.10.254

Starting Nmap 7.12 ( https://nmap.org ) at 2016-07-18 22:21 PDT
Nmap scan report for 10.10.10.254
Host is up (0.0022s latency).
PORT   STATE  SERVICE
22/tcp closed ssh
23/tcp open   telnet

Nmap done: 1 IP address (1 host up) scanned in 13.03 seconds
Trying 10.10.10.254...
Connected to 10.10.10.254.
Escape character is '^]'.

TM02 login: admin
Password:
login: can't chdir to home directory '/data/UsbDisk1/Volume1'
$ echo $USER
admin
$ cat /etc/shadow
root:$1$D0o034Sm$LY0jyeFPifEXVmdgUfSEj/:15386:0:99999:7:::
bin:*:13341:0:99999:7:::
daemon:*:13341:0:99999:7:::
```

A better way?

- Pretty much all run OpenWRT
- They're REALLY AWESOME for price
  - ~~$30~~ $25 gets you either:
- 256mb of RAM, 500mhz processor, and 64mb of flash, microSD Slot
- 64mb of RAM, 400mhz processor, 16mb of flash, PoE
- Pretty sweet specs for a cheap device that fits in your palm
- Time to put them to use!

One small problem: Value Add

```
model=$(awk 'BEGIN{FS="-"} /machine/ {print tolower($2)}' /proc/cpuinfo)
if [ "$model" = "connect inet v1" ]; then
        model="6416"
        ddns=$(dd if=/dev/mtd0 bs=1 skip=$((0x1fc10)) count=7 2>/dev/null)
        code=$(dd if=/dev/mtd0 bs=1 skip=$((0x1fc20)) count=16 2>/dev/null)
elif [ "$model" = "ar150" -o "$model" = "ar300" ]; then
        ddns=$(dd if=/dev/mtd6 bs=1 skip=$((0x10)) count=7 2>/dev/null)
        code=$(dd if=/dev/mtd6 bs=1 skip=$((0x20)) count=16 2>/dev/null)
elif [ "$model" = "mt300a" -o "$model" = "mt300n" -o "$model" = "mt750" ]; then
        ddns=$(dd if=/dev/mtd2 bs=1 skip=$((0x4010)) count=7 2>/dev/null)
        code=$(dd if=/dev/mtd2 bs=1 skip=$((0x4020)) count=16 2>/dev/null)
fi
```

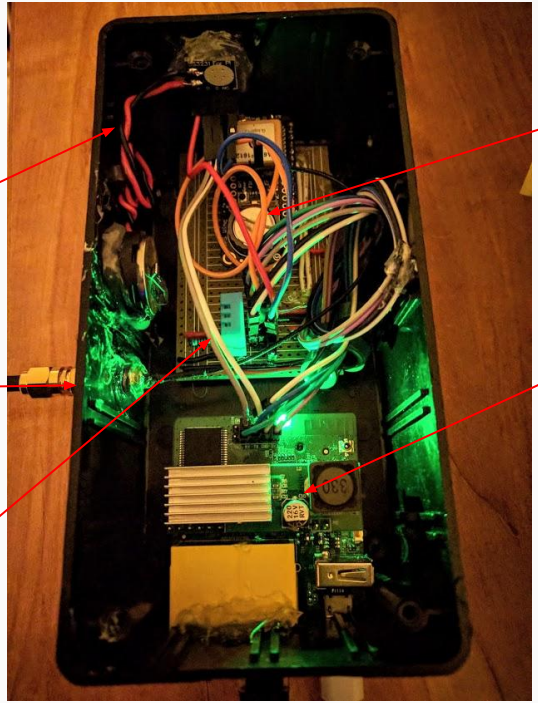
```
if [ "$has_new" = "yes" ]; then
        echo "found new version, will download now"
        wget "http://www.gl-inet.com/firmware/$model/v1/$file" -O /tmp/auto_firmware.img
        #after download, do a md5 check
        md5=$(md5sum /tmp/auto_firmware.img |awk '{print $1}')
        if [ "$md5" != "$md5_check" ]; then
                rm -fr /tmp/auto_firmware.img
        else
                #set to check per hour wait for 4:00am and continue to next cycle
                interval=$one_hour
                continue
        fi
fi
```

Stratum-1 GPS NTP Server

- High Accuracy
- No need to connect to the internet
- Self contained and very low power!
  - ~300 mA/h
  - PoE Capable

- Gl.iNet AR150
  - 400mhz
  - 16MB ROM / 64MB RAM
  - 4 pins GPIO

Final Result:



Getting there...

- We need:
  - Serial to be free (for GPS to use)
  - PPS via GPIO (Pulse Per Second)
  - Easy deployment
  - i2C Support and DHT Support

Building Made Easy

- **Tips:**
  - Make menuconfig - good for configuring packages, resources, and anything “optional”
  - Make kernel_menuconfig - Internal modules built into the kernel - RTC, PPS, GPIO modules are here.
  - When done, always make defconfig

```
.config - Linux/mips 4.1.35 Kernel Configuration
                    Linux/mips 4.1.35 Kernel Configuration
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes
features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] built-in
[ ] excluded  <M> module  < > module capable

        Machine selection  --->
    [*] OpenWrt specific image command line hack
        Endianness selection (Big endian)  --->
        CPU selection  --->
        Kernel type  --->
        General setup  --->
    [ ] Enable loadable module support  --->
    [*] Enable the block layer  --->
        Bus options (PCI, PCMCIA, EISA, ISA, TC)  --->
        Executable file formats  --->
        Power management options  --->
        CPU Power Management  --->
    [*] Networking support  --->
        Device Drivers  --->
        Firmware Drivers  --->
        File systems  --->
        Kernel hacking  --->
        Security options  --->
    -*- Cryptographic API  --->
        Library routines  --->

            <Select>    < Exit >    < Help >    < Save >    < Load >
```

Building Made Easy

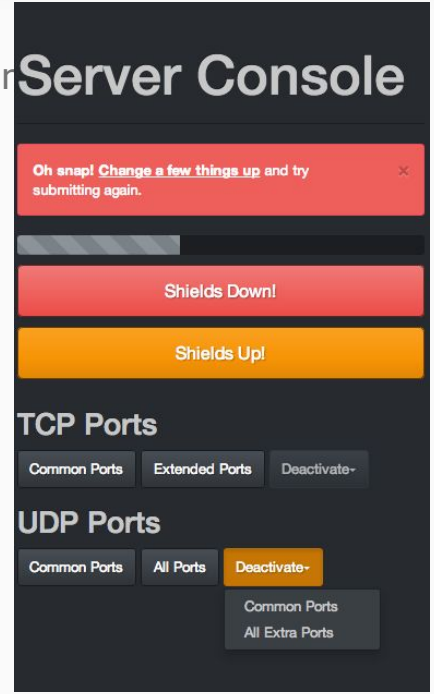
- **Files:**
  - Full root structure in ./files/
  - Configurations:
    - Rc.local - Runs at boot, good for some settings
    - Init Scripts - Better, runs at specific target
    - Inittab - By default responds on serial interfaces

What to include?

- Chrony has built in support for RTCs and PPS
- GPIO-PPS
- Lsof
- NTP Utils
- GPSD
- Custom GPIO-PPS “driver”
    - By default driver has no settings
    - You must write mappings to support each device IO type
    - AR7XXX has IRQ so we can use that

Why?

- ImageBuilder / Source is significantly smaller than adding packages after in
- Allows us to deploy settings, configurations, again and again
  - Mesh networks
  - Cheap APs
  - Easy restore
- My current uses:
  - Low Power Emergency Box
  - NTP Server
  - Travel Hotspot/Router
  - Network Tap

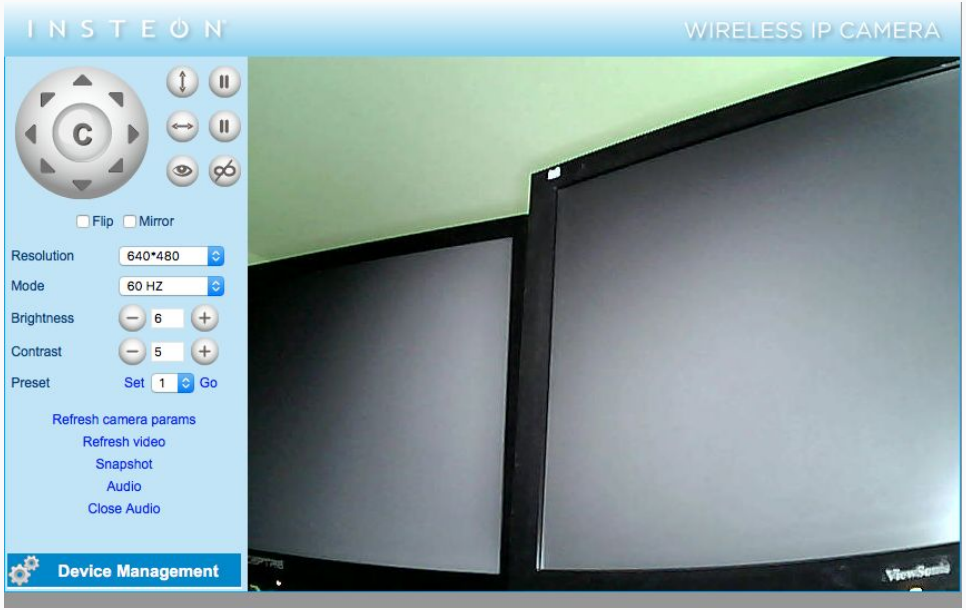


Time to build something!

Fosscam (Clones)

- Runs Linux 2.4-uc0
- Very modern with full IPv4 networking stack!
- Not a lot of space to customize, but easily accessible serial
- Some clones are implemented poorly, have vulnerabilities and telnet
- Some clones can swap firmware with other manufacturers

- API is based on a SDK
- We can use this to connect and use the camera features



TP-Link HS100

- Like other “Smart Plugs” has no authentication
- Designed to be used “locally” or “in the cloud”
- Protocol is just static-key rotation, easy json on decode
- No obvious way to reflash (unlike WeMo), UART accessible

- **Not ideal But: Just put on its own its own WiFi**
- **How to use it though?**

Smart WiFi

- Now have an isolated network, but how do we use it?

Light Dude

Light Dude

- Amazon Dash Buttons are fun
  - Connects to WiFi
  - Uses AA battery to power SOC
  - Very low power
- Performs DHCP request and TLS connection to Amazon
  - We can listen to DHCP
  - Sadly it makes multiple requests…

```
Mar  2 04:12:04 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
but message was ignored due to recent trigger (Last trigger: 4 seconds ago.)
Mar  2 04:12:13 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
but message was ignored due to recent trigger (Last trigger: 13 seconds ago.)
Mar  2 05:48:14 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
and triggering lights!
Mar  2 05:48:14 carbon lightdooder[1219]: Running toggle on light switch 192.168.2.22 - setting
 state to 'on'
Mar  2 05:48:14 carbon lightdooder[1219]: Running toggle on light switch 192.168.2.21 - setting
 state to 'on'
Mar  2 05:48:19 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
but message was ignored due to recent trigger (Last trigger: 4 seconds ago.)
Mar  2 05:48:28 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
but message was ignored due to recent trigger (Last trigger: 13 seconds ago.)
Mar  2 05:49:07 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
but message was ignored due to recent trigger (Last trigger: 52 seconds ago.)
Mar  2 05:49:12 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
but message was ignored due to recent trigger (Last trigger: 57 seconds ago.)
Mar  2 05:49:21 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
and triggering lights!
Mar  2 05:49:21 carbon lightdooder[1219]: Running toggle on light switch 192.168.2.22 - setting
 state to 'off'
Mar  2 05:49:21 carbon lightdooder[1219]: Running toggle on light switch 192.168.2.21 - setting
 state to 'off'
Mar  2 17:58:42 carbon lightdooder[1219]: Recieved Message from Dash Button: xx:xx:xx:xx:xx:xx
and triggering lights!
Mar  2 17:58:42 carbon lightdooder[1219]: Running toggle on light switch 192.168.2.22 - setting
 state to 'on'
Mar  2 17:58:42 carbon lightdooder[1219]: Running toggle on light switch 192.168.2.21 - setting
 state to 'on'
```

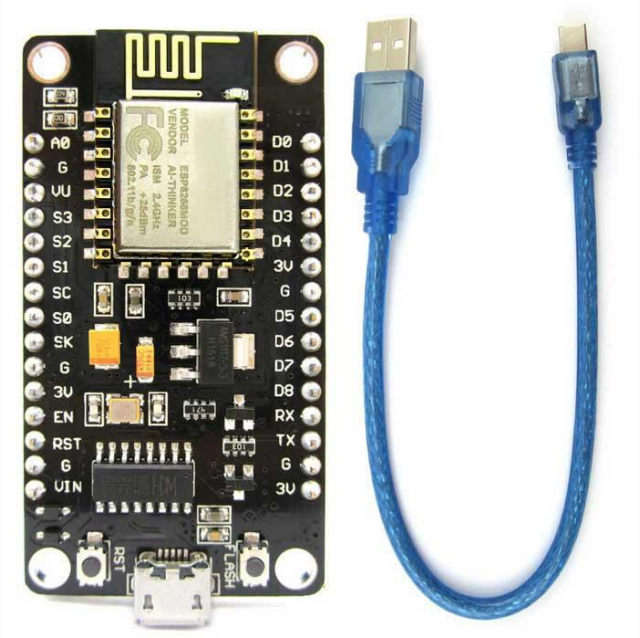
Light Duder

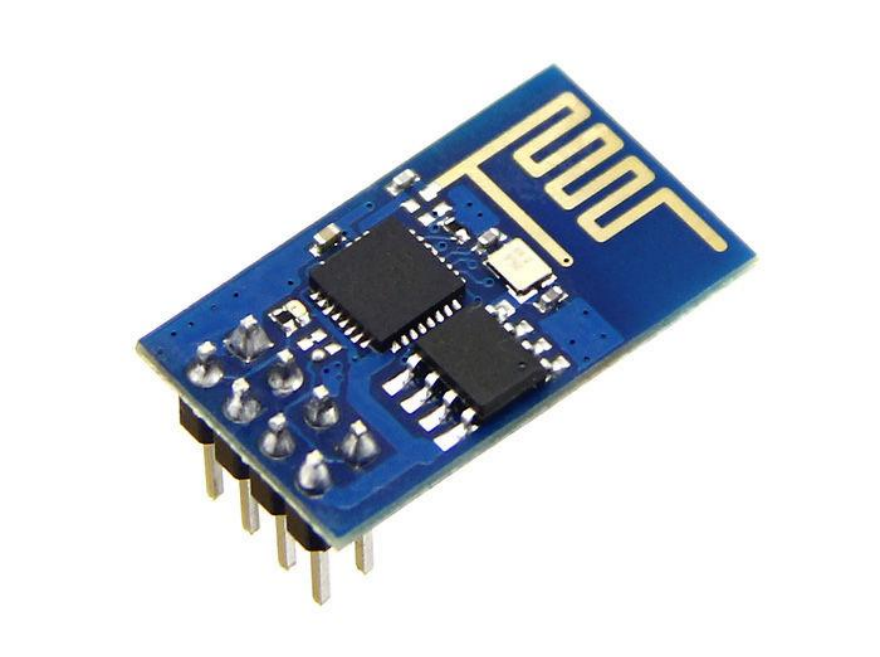
- Taking multiple IoT devices and using them for good!
  - Smart Camera (From before)
  - Amazon Dash Buttons
  - Real Time / Sunrise / Sunset Data
- Automatically turn on lights when:
  - motion is detected
  - Multiple rules trigger
  - Sunrise/Sunset
  - Weather

Light Duder


SOUNDS GOOD TO ME!

One last Note





Great! But...

- I actually have a hybrid of these suggestions
- I have a bridge router to connect my network and the IoT
  - Allows access to weather reports
  - Allows access to syslog (out)

This allows me to keep the risk relatively low but provide all the features I need without the IFTTT / Internet

Any questions?

Feel free to contact me:

On Twitter: @spiceywasabi