

A vintage television set with a wooden-grain frame and two control knobs on the right side. The screen displays a red and black splatter pattern. The TV is tilted and positioned in the lower-left corner of the slide.

SmartTV ~~IN~~Security

ReVuln Ltd.

PhDays 2014

Who?



Donato Ferrante
@dntbug



Luigi Auriemma
@luigi_auriemma



ReVuln?



Vulnerability Research

Penetration Testing

Consulting

revuln.com

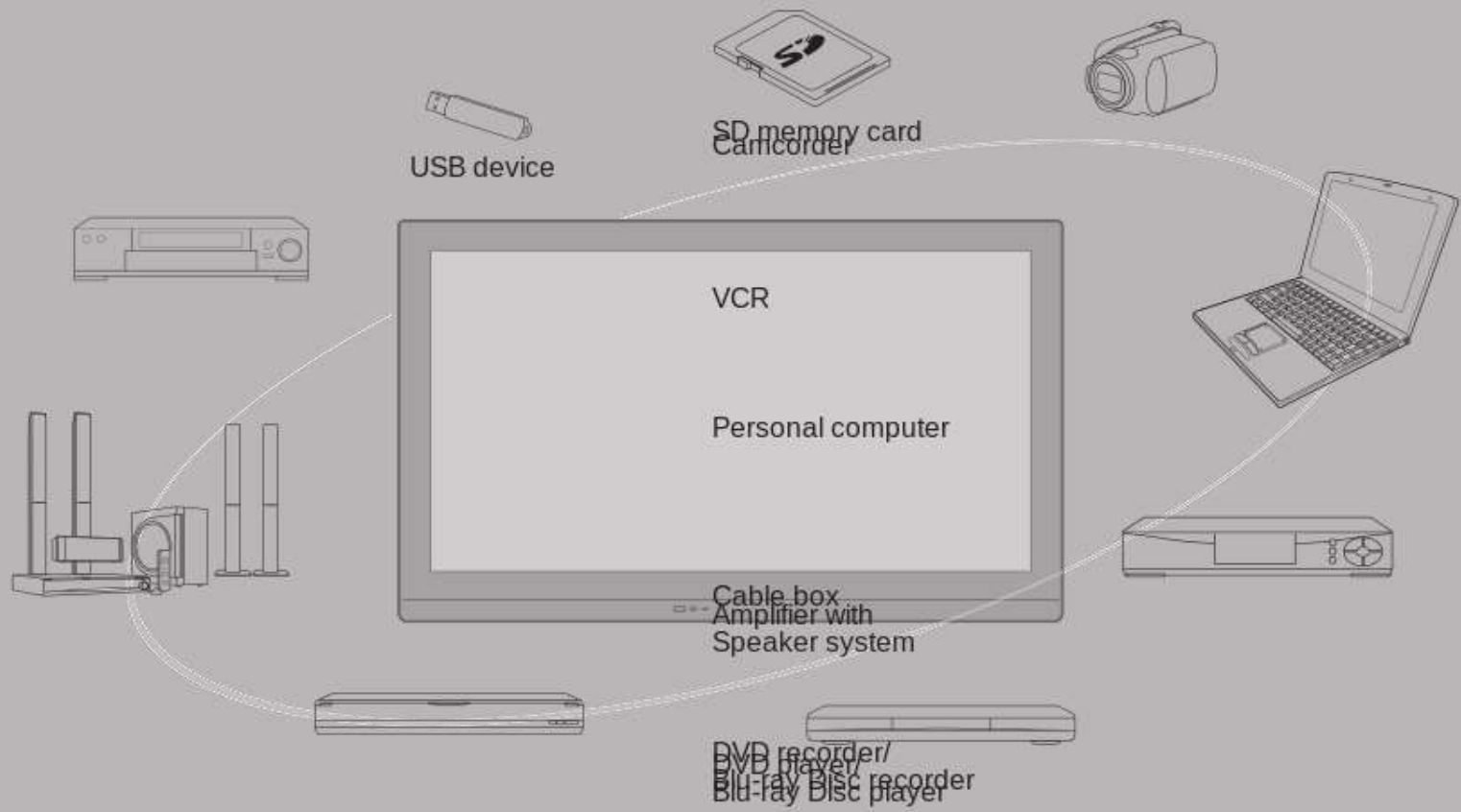
info@revuln.com

@revuln

What's a SmartTV? (1)

- Describes a trend of **integration of the Internet and Web 2.0 features into television**
- Technological convergence between computers and television sets and set-top boxes
- SmartTV = **a television with integrated Internet capabilities** that offers more advanced computing ability and connectivity than a contemporary TV

What's a SmartTV? (2)





Not all the TVs are SmartTVs...



but a **lot** of TVs are actually SmartTVs



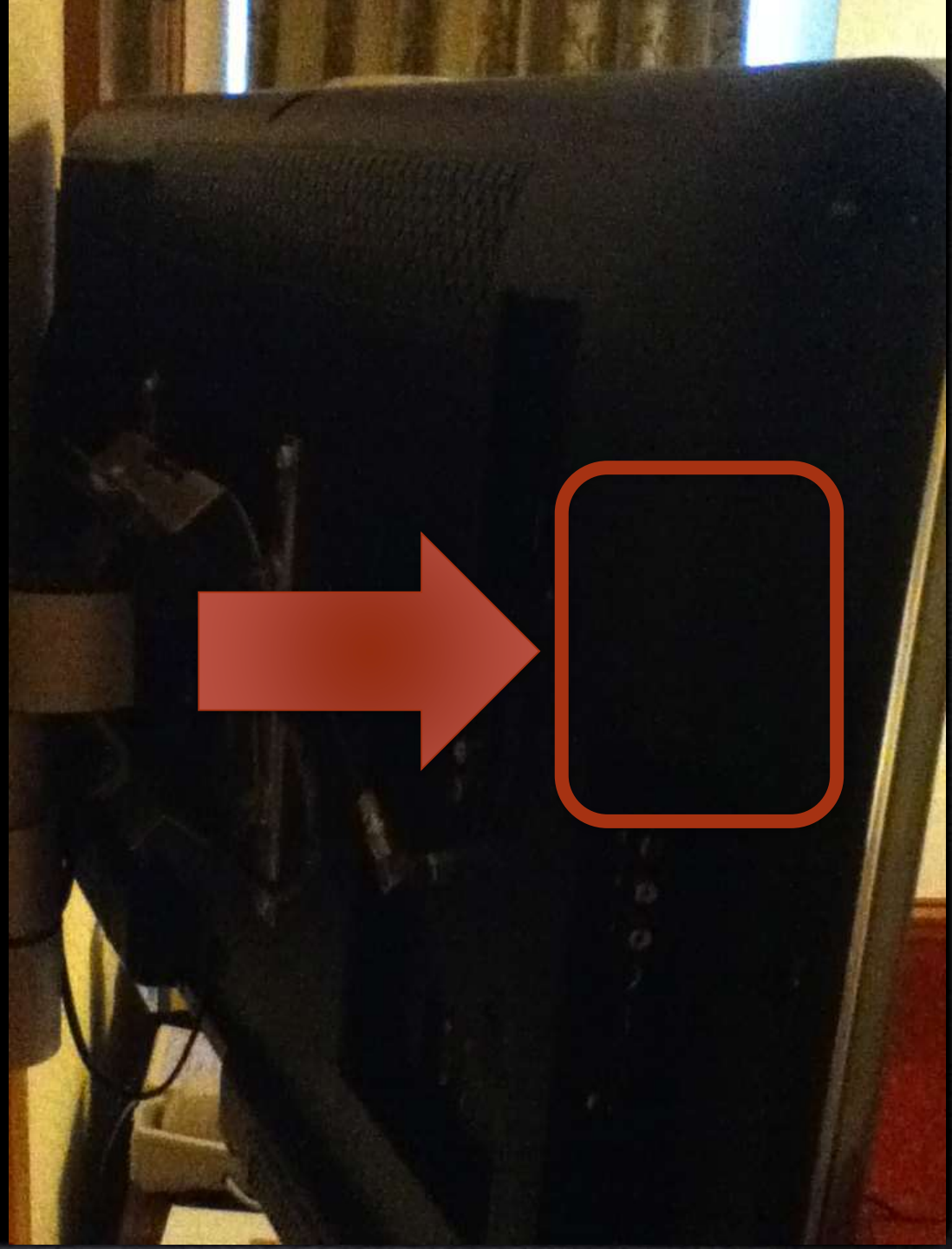
SmartTVs at the Airport..

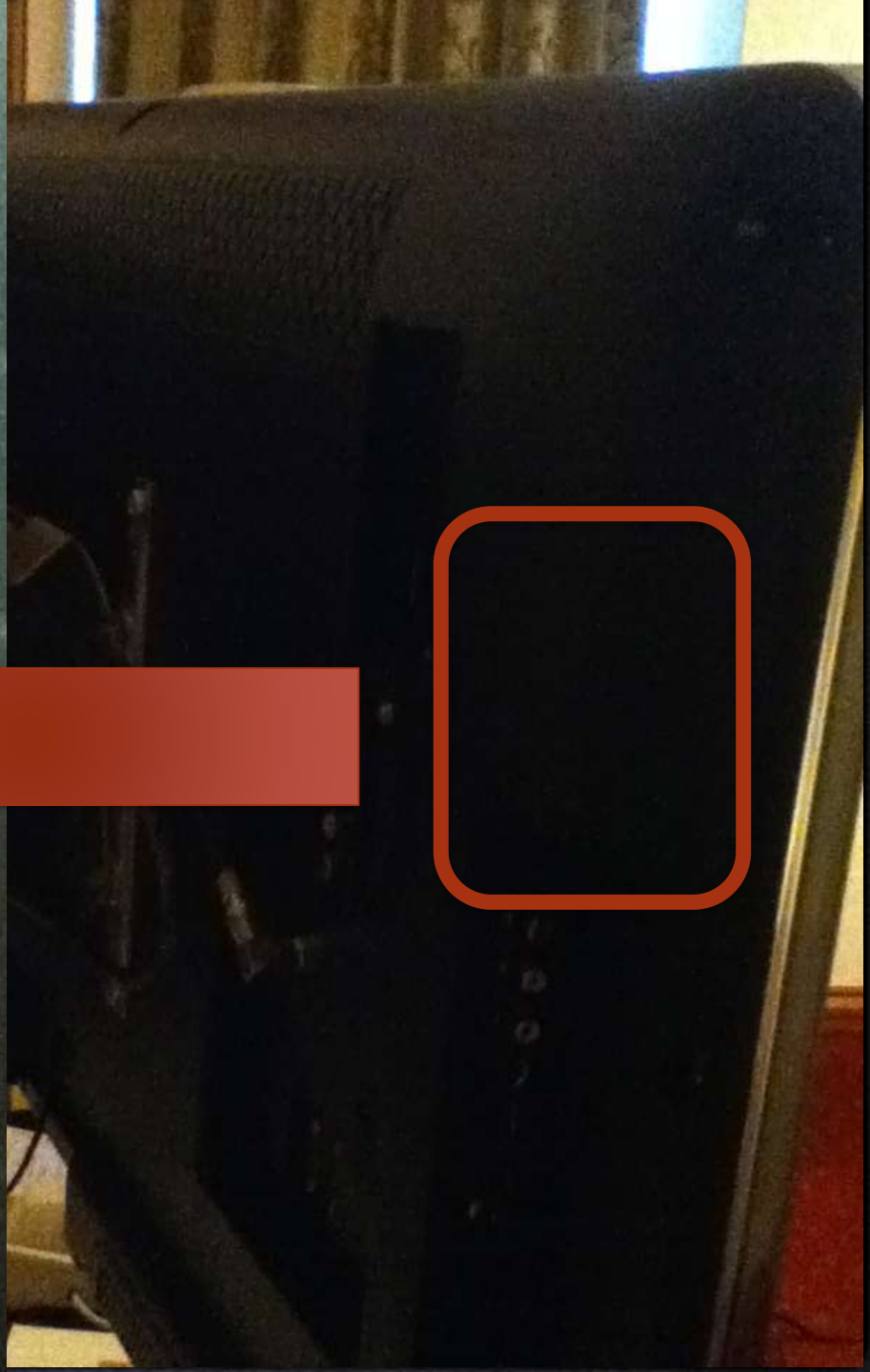
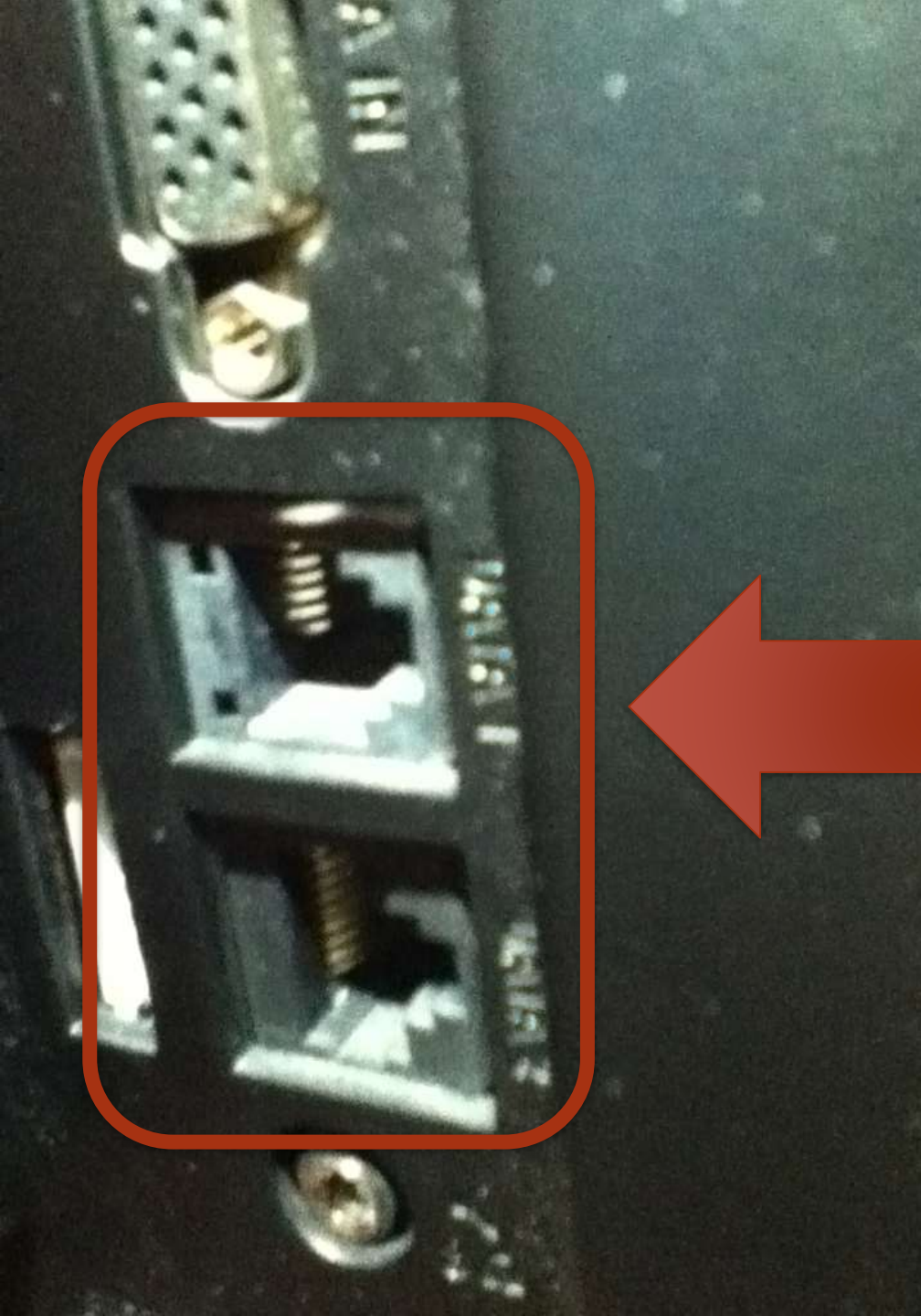


And in our **hotel** room..
SmartTV or **not** SmartTV, that is the **problem**..

We checked the back of the TV and it was dark..

So we turned on a **flashlight** and..



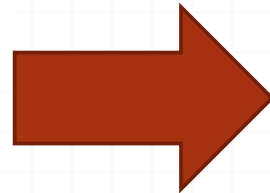


Before and After



TV

Get input signal then output



SmartTV

Fully-featured PC

Why are SmartTV so popular?



1st Commercial

- If you have to choose between a simple TV and a TV with a lot of features, even features that you don't know (but they sound cool), you will go for the second one



2nd Advertising

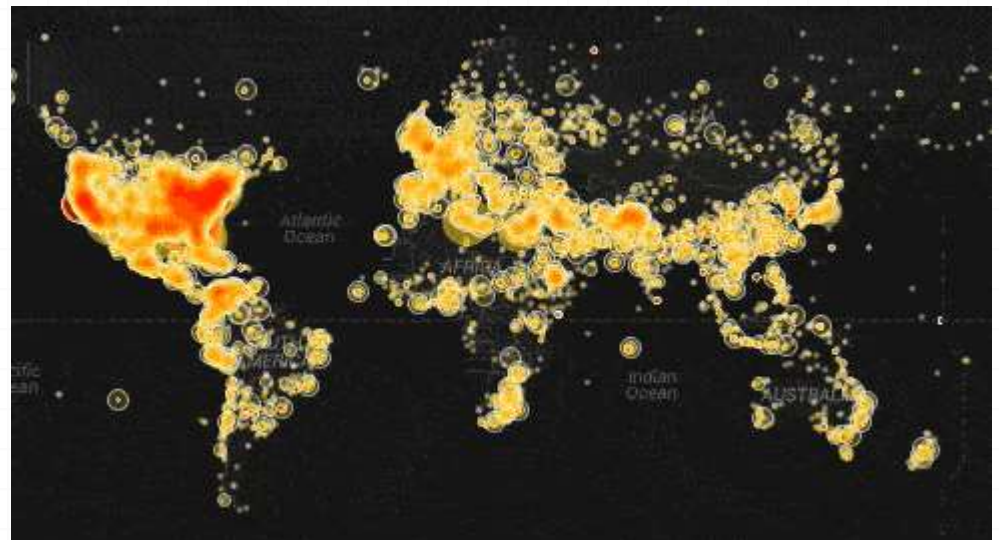
- Advertising = Money for Vendors/Ads Providers
- Targeted advertising and other advanced advertising features such as ad telescoping using **VOD** and **PVR**, enhanced TV for consumer call-to-action and **audience measurement** solutions for ad campaign effectiveness
- **Bidirectional flow** between TV and Ads providers

Advertising and Security

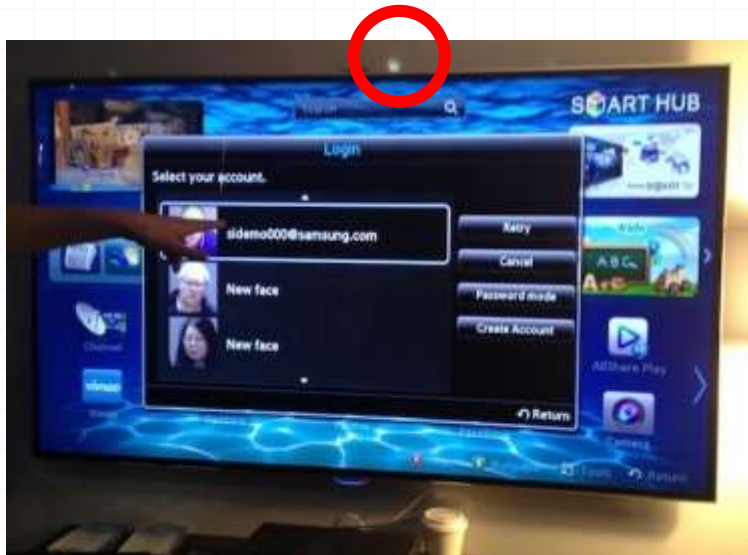
- o This bidirectional flow between TV and Ads provider, has 2 main consequences:
 - o **Privacy**, the viewer is disclosing personal habits
 - o **Security**, a man-in-the-middle attack can be pretty effective to achieve one of the following goals:
 - o **Ads-Hijacking** => To influence the viewer
 - o **Vulnerability Exploitation** => To get access to the TV

Why SmartTV as Target?

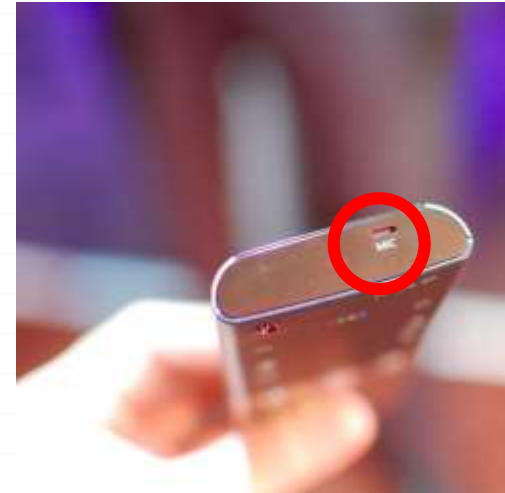
- o Used worldwide
- o Huge attack surface
- o And..



TV can see you..



TV can hear you..



BIG BROTHER



IS WATCHING YOU

1984

- o An attacker able to gain access to your SmartTV can:
 - o Get access to your **Home** privacy
 - o Get access to your **Company** meeting room
 - o And more..

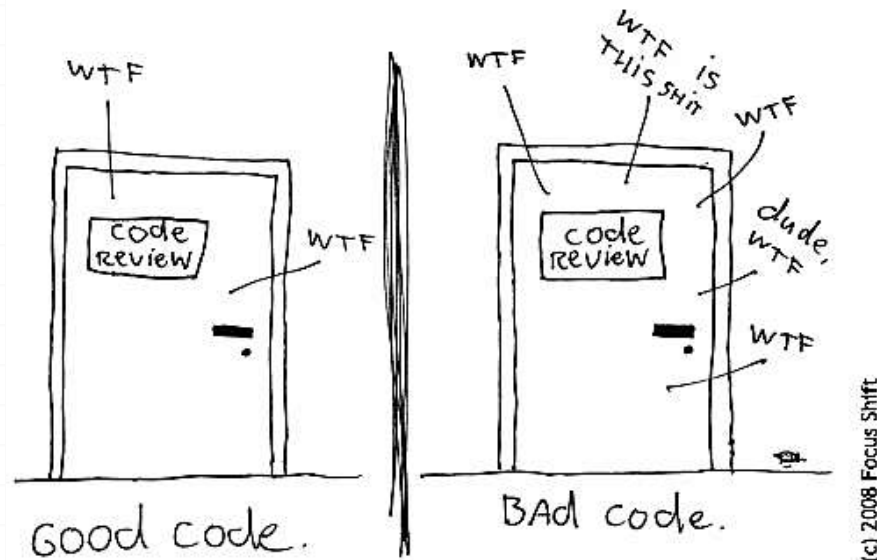
The Problem (1): Insecurity



o **SmartTV are insecure!**

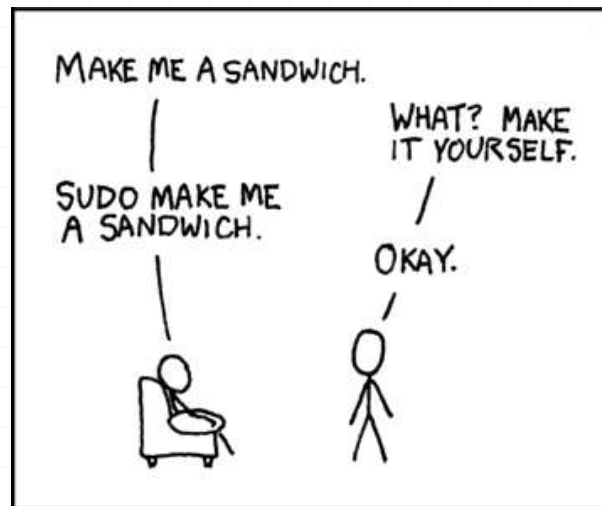
The Problem (2): Bad & Obscure

The ONLY VALID MEASUREMENT
OF CODE QUALITY: WTFs/MINUTE



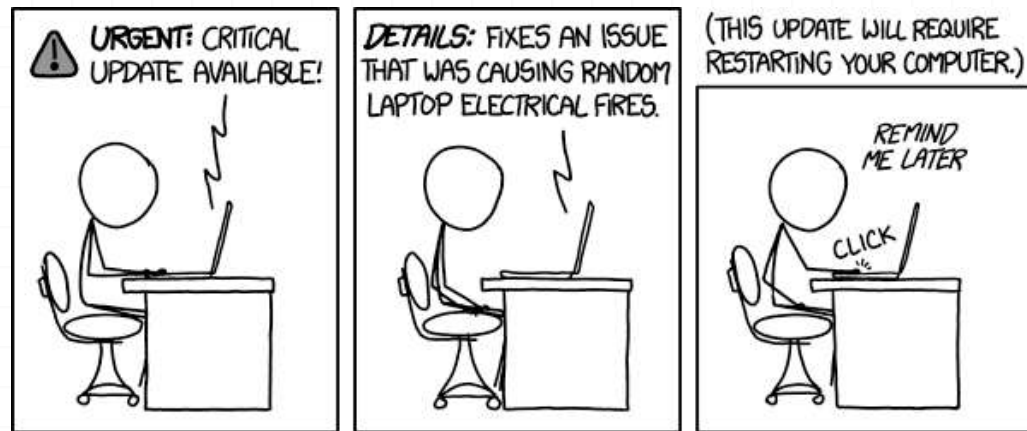
- They rely on **bad coding practice** and..
- They usually rely on **security by obscurity** (sigh)

The Problem (3): Server & Client



- o Affected both by **Server-side issues**.
 - o There are several **services** running and listening for incoming connections (by default)
- o And **Client-side issues**
 - o Any of the **Apps** installed by default can represent a possible attack vector against the device itself

The Problem (4): Updates



- o A lot of **software installed** on the TV..
- o Have you **ever updated** your TV?
- o How **security fix** are **pushed** on your TV by the Vendor?
- o Are you running the **latest release** of the **web browser**?

Nice..

But now tell us how to get a \$hell!



Things to know.. (1)

- A SmartTV is an expensive hardware device
 - Usually > 1000 Euro (47.000 RUB)
- You might “brick” the TV (no longer works)
- Big hardware and software differences between the TV models, even those of the same vendor
- Multiple names for the same features (i.e. HDMI-CEC*)

* <http://en.wikipedia.org/wiki/HDMI#CEC>

Things to know.. (2)

- SmartTV are usually based on Linux
- Using MIPS and ARM CPU
- Having a number of different embedded stuff including: WiFi, USB, Camera, Microphone, sensors, etc
- Running a wide range of proprietary and customized software, with crazy configurations
- Black-box testing means wasting lot of time to get information, having few control over the TV and limited debugging

How to get the Software? (1)

- o SmartTV vendors (like Samsung and LG) usually release emulators and/or SDK for developers willing to create new Apps for the TV
- o The idea of using the emulators on the PC to find issues affecting the TV might sound interesting
- o The problem is that the emulator doesn't usually match the software running on the real TV
- o For example if you find 10 issues in the emulator, probably only 1 or 2 will work on the TV and bugs affecting the TV may not work on the emulators
- o But emulators are good to have an idea of some protocols and how the code works

How to get the Software? (2)

- Via firmware updates

 - Don't need to access the TV

 - Thousands of updates available for free on the Vendors websites

 - Usually encrypted with an encryption key defined on a TV/model base. i.e.: 2 different models of the same vendor will have 2 different keys

 - Require some reverse engineering work to extract the content

How to get the Software? (3)

- Via directory traversal
 - Needs a vulnerability
 - If you can access /proc you have lot of information
 - If you can access /dev you can download all the filesystems
 - Otherwise you have to guess file/directory names by using some techniques

How to get the Software? (4)

- Via code execution
 - Needs a vulnerability
 - Full access to files, directory and attached devices!
 - Execute whatever commands you want :]
 - **Bye Bye TV Caveat:** You might brick the TV!!!

How to get the Software? (5)

- Via JTAG or NAND/SD physical reading
 - Hardware solution, you must open the TV and playing with its content... bye bye warranty
 - Not always available or easy to access
 - It might cause some trouble to the device
 - A lot of effort and only for skilled people

Reset: Service Menu

```

Panel On Time(Hour) 00119
1. Calibration
2. Option Table(Service)
3. White Balance
4. SVP-UX
5. Option Block
6. SGTV5810/NTP3000
T-BOPMPELD-1002 Boot Merge : OK
T-BOPMPELIS-1002
BORD2_CALLA_TR-1002
Feb 28 2007
08:20:16
BORDEAUX+ ExtL2
7. YC Delay
8. Adjust
9. I2C Check
10. W/B MOVIE
11. Checksum
12. Reset
13. Spread Spectrum
    
```



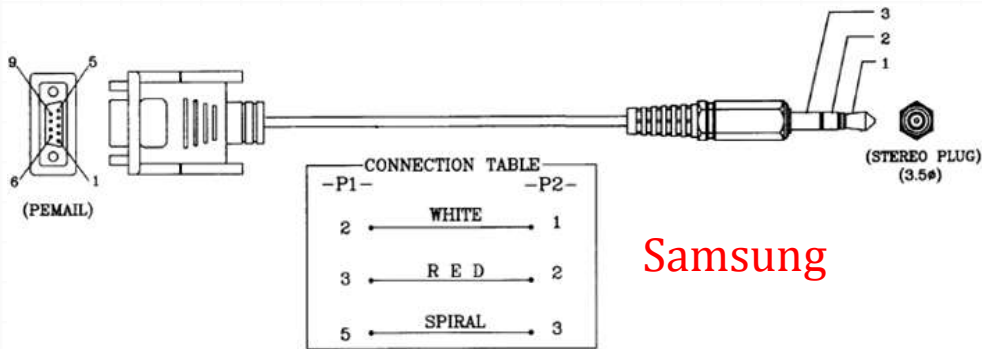
```

DIGITAL SERVICE
001 OP
000 VERS

CHANO (EXT)
DMG.211EDA HF:3,4,6,1024
HF3.001M00AA HF:-----
DF3.052M00AA FD:-----
YH1.008M00AA BT:-----
HE.001C 1,2,14,000
CMG.200EUA)
DQG.120M00AA <FS>
PK3.120M00AA EFR:03,09,00,10
MGL.211MM SMC:-----
MID:07802AB1 SW1:-----
PDI:0E000000 SW2:-----
PK1:0E000000 TSC:-----TSC:-----
PK1 SCF:20,07
RESAQV002,0 SYF:-----
    
```

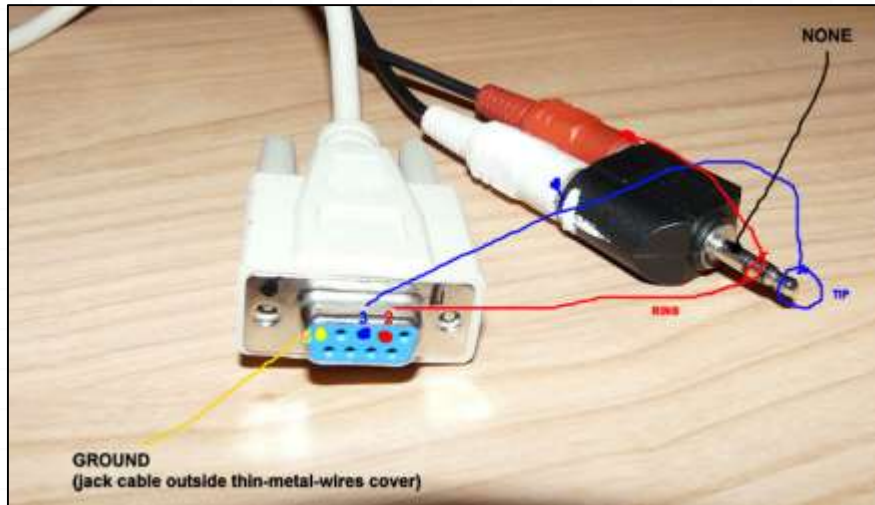


Debug: Serial Cable



Samsung

Philips



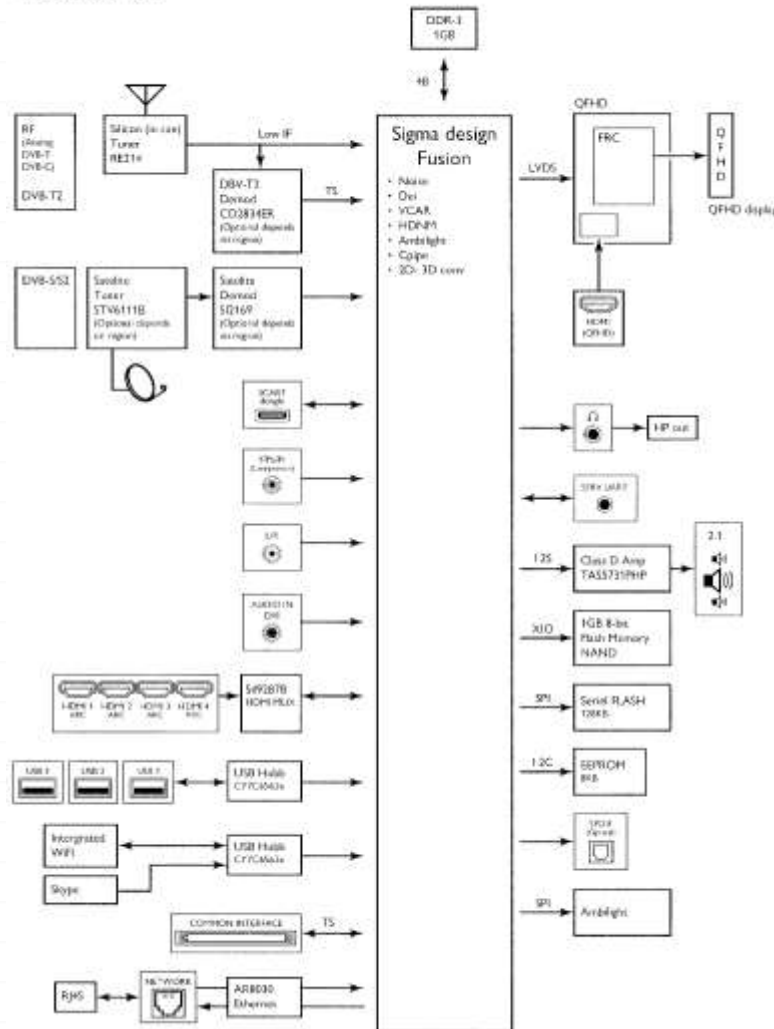
LG

Attack Surface

- o As you might have guessed there are a lot of different ways to attack a SmartTV
- o To get a better understanding let's take a look at a real world device
- o We will just focus on a subset of the device attack surface
- o To do that we take in consideration the following schema related to a Philips SmartTV...

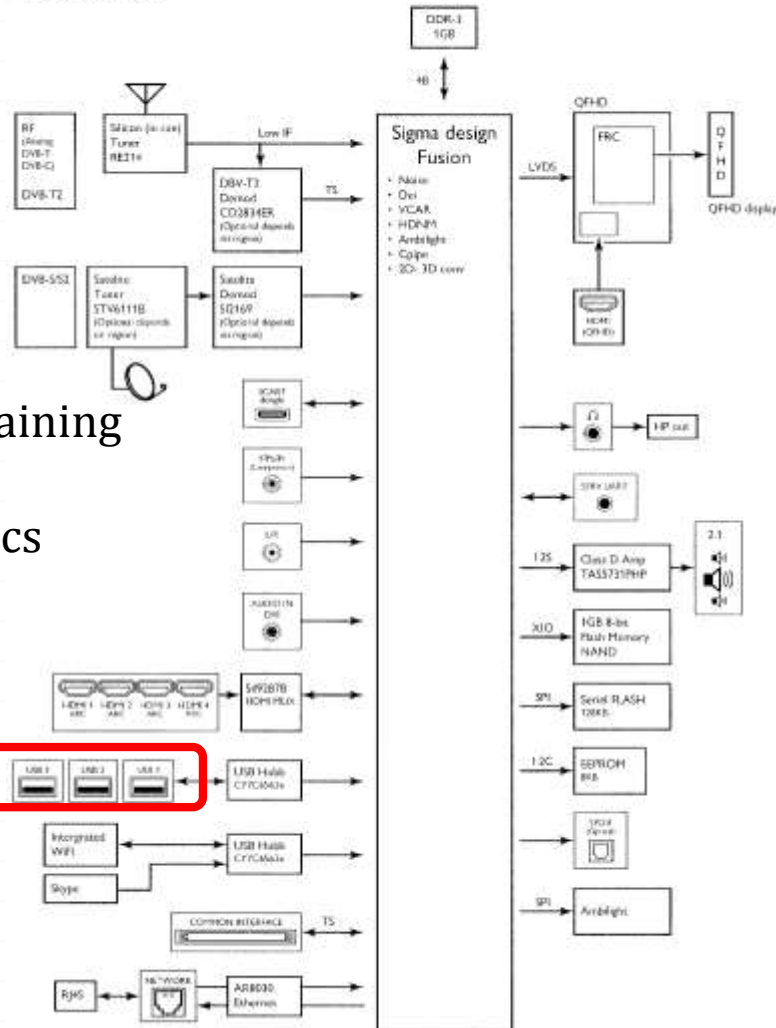
Attack Surface example from Philips manual

9000 series



Attack Surface - USB

9000 series



Malicious USB stick containing malformed data i.e.:

- Video and Audio codecs
- Filesystem
- USB stack
- Auto executed files



Attack Surface - HDMI

9000 series

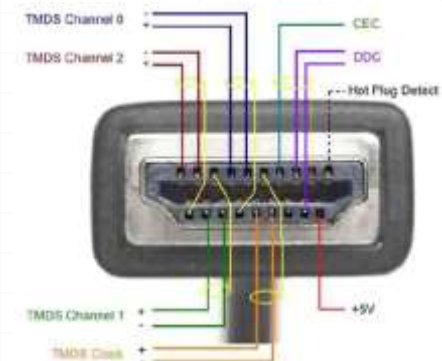
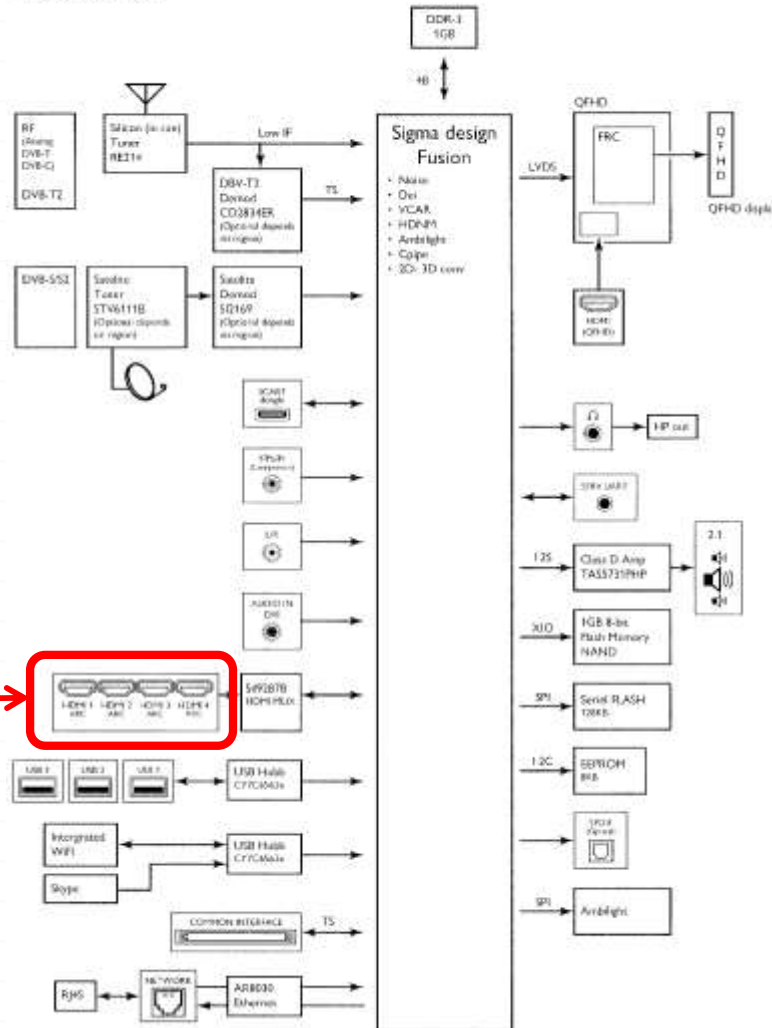
Communication protocols:

- CEC
- HEC*

for device interoperability

Rogue hardware via Ethernet connection (HEC)

HDMI



*HEC is not that popular, not clear how many devices are using this standard..

Attack Surface - DVB

9000 series

DVB

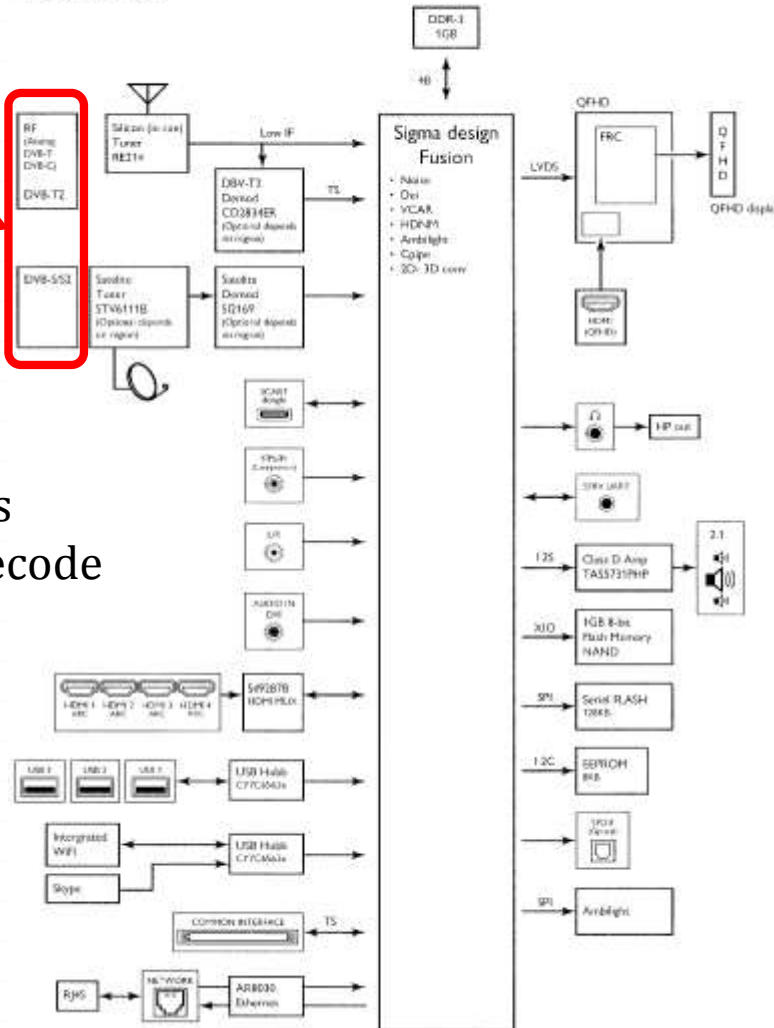
Radio signal to the TV

DVB != Analog

It's a protocol, which allows using different codecs to decode the video/audio streams

Different standards:

- DVB-T (terrestrial)
- DVB-C (cable)
- DVB-S (satellite)



How to make a DVB-T Pirate Channel – COFDM modulator transmitter generator for HDMI / CVBS video source



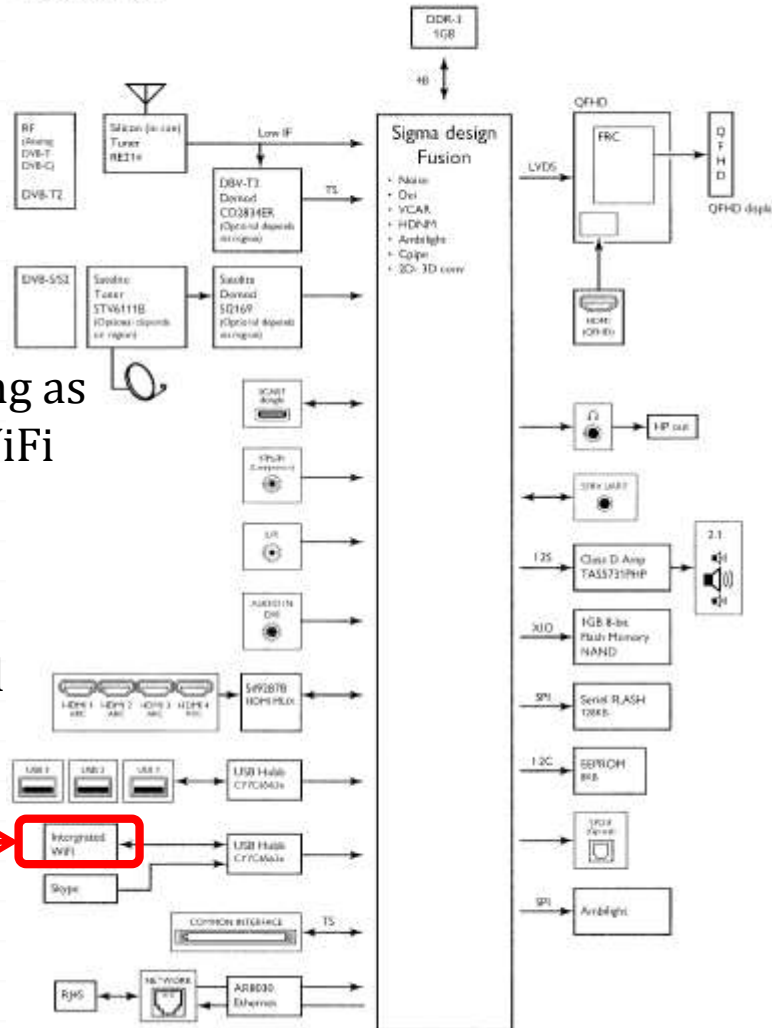
Homemade DVB transmitter

Wait! Before we forget..

- o The DVB audio/video streams are a possible fuzzing target:
 - o HEVC, H.262, H.264, AVS, MP2, MP3, AC-3, AAC, HE-AAC
- o But the embedded interactive content is the best way to attack the TV:
 - o HbbTV
 - o CE-HTML
 - o MHEG

Attack Surface - WiFi

9000 series



WiFi adapter of the TV acting as access-point listening for WiFi connections.

The **Miracast** protocol is composed by out-of-band WiFi packets, protocols and codecs

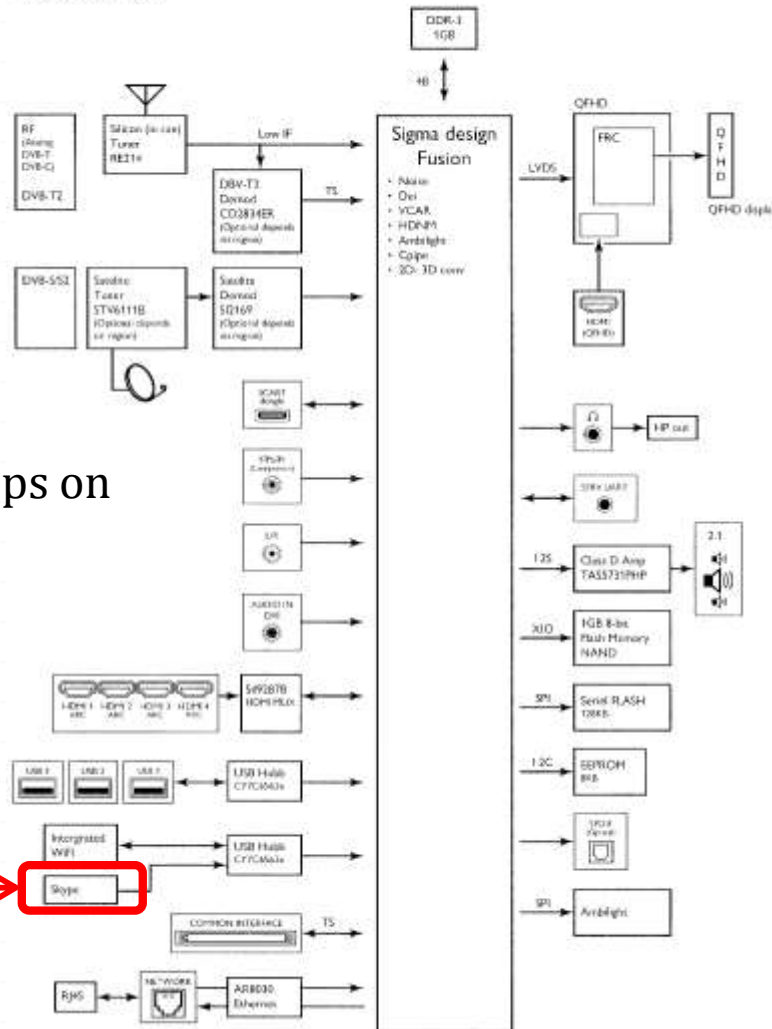
WiFi → Integrated WiFi



A vulnerability in Miracast allows the attacker to access the TV from outside your house

Attack Surface - Apps

9000 series



Vulnerabilities affecting Apps on the TV:

- Skype
- Web browser
- Malicious Apps

Apps → **Skype**



Attack Surface - LAN

- o Most of the SmartTV issues are related to services exposed via LAN:
 - o UPNP
 - o Video/Audio service (like DirectFB)
 - o Various HTTP/HTTPS servers
 - o Network remote controller
 - o Media sharing
 - o Status and information
- o First thing to try on your SmartTV is using NMAP:
 - o You will see a number of different TCP and UDP ports open
 - o Some of them awaiting for you to connect :]
 - o If you try to send some junk data to these ports you might get some easy way to crash/reboot the TV, a starting point to investigate
- o The TV also scans the LAN, an attacker can passively exploit vulnerabilities

Real World Issues

The TV is Watching You

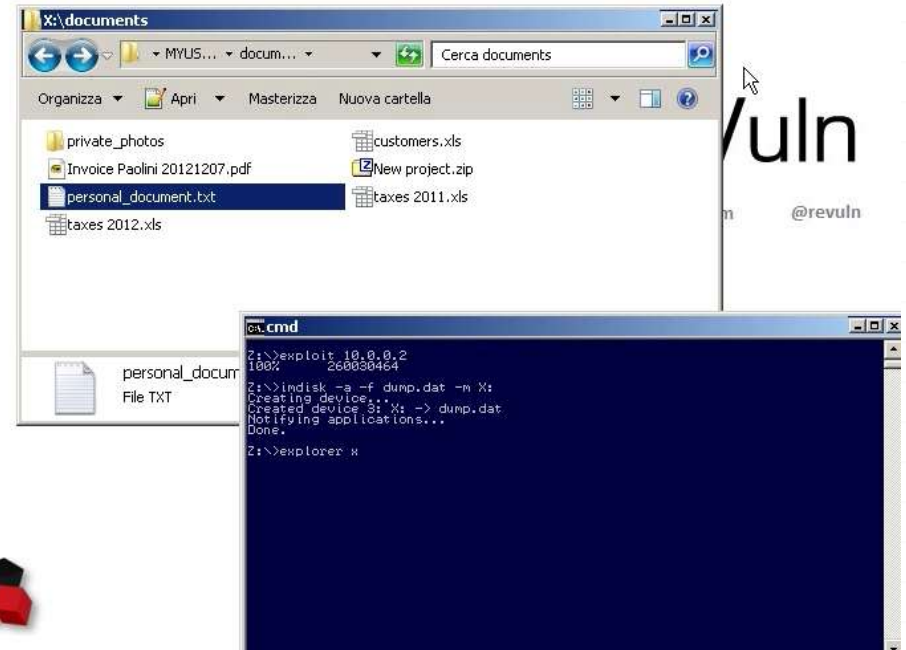


Samsung #1 (1)

- o **Date:** 2012
- o **Tested device:** Samsung SmartTV D6000
- o **Affected Service/Protocol:** DMRND, an HTTP server listening on ports 52253 and 52396
- o **Vulnerability:** Directory Traversal, which allows to download any file available on the device except special files like those in /proc
- o **Details:** The server has a security check to allow people to download files having only whitelisted file extensions (jpg, png, ..). To bypass the check the attacker needs to append a NULL byte followed by the whitelisted extension:
 - o <http://SERVER:52235/../../../../etc/passwd%00.png>

Samsung #1 (2)

- o Download all the filesystems from the remote TV
- o Download the filesystems related to all the connected USB devices



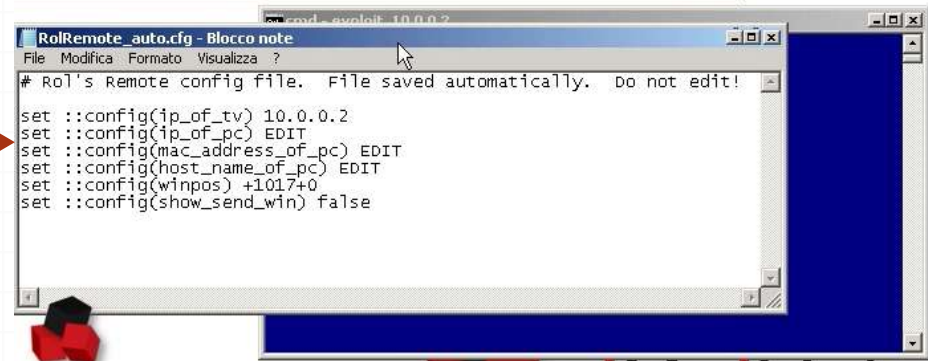
Samsung #1 (3)

TV controller configuration file, it contains the parameters used by the whitelisted remote controller →



```
mp.dat - Blocco note
Modifica Formato Visualizza ?
.-><devname>mycontroller<devname><ip>1.2.3.4<ip><mac>11-22-33-44-55-
```

Configuration file used by the our PC program, we need only to copy the above parameters here →

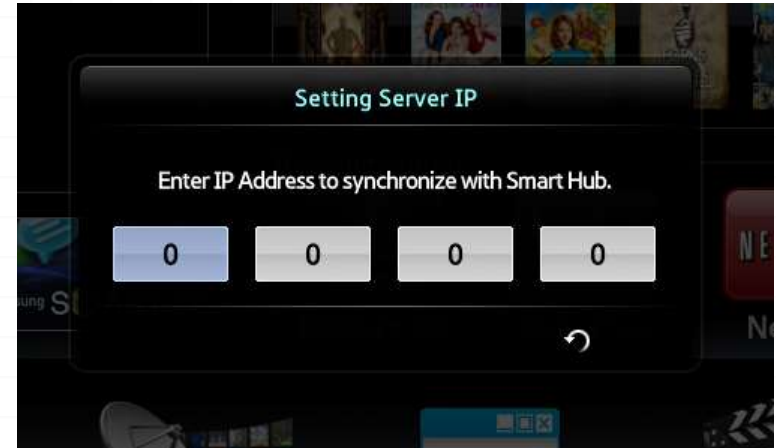


```
RolRemote_auto.cfg - Blocco note
File Modifica Formato Visualizza ?
# Rol's Remote config file. File saved automatically. Do not edit!
set ::config(ip_of_tv) 10.0.0.2
set ::config(ip_of_pc) EDIT
set ::config(mac_address_of_pc) EDIT
set ::config(host_name_of_pc) EDIT
set ::config(winpos) +1017+0
set ::config(show_send_win) false
```

These fields are not part of the Ethernet packets, but are defined inside the protocol itself so it's possible to spoof the IP, MAC address and hostname to allow an attacker in the network to impersonate the whitelisted TV controller

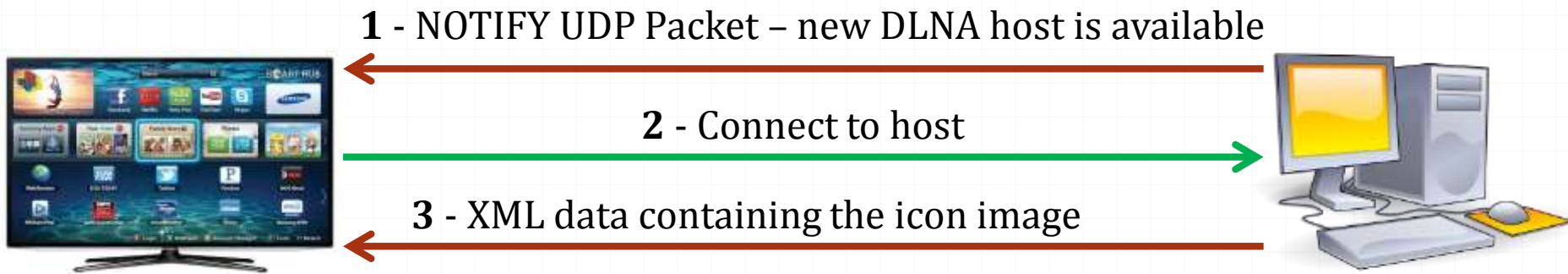
Samsung #1 (4)

- o Now we can control the TV when the victim is not watching
- o The attacker can install arbitrary malicious Apps on the TV using the “develop” account



Samsung #2 (1)

- o **Date:** 2012
- o **Tested device:** Samsung SmartTV D6000
- o **Affected Service/Protocol:** DLNA client
- o **Vulnerability:** Buffer overflow exploitable as soon as the device tries to download the icon image associated to a DLNA host



Samsung #2 (2)

1 - NOTIFY UDP Packet – new DLNA host is available



```
NOTIFY * HTTP/1.1
Host: 239.255.255.250:1900
Location: http://192.168.0.3:56923/test.xml
NTS: ssdp:alive
Cache-Control: max-age=1800
Server: UPnP/1.0 DLNADOC/1.50 Platinum/0.6.8.0-bb
USN: uuid:00000000-0000-0000-0000-0000-000000000000::urn:schemas-upnp-org:device:MediaServer:1
NT: urn:schemas-upnp-org:device:MediaServer:1
```

Samsung #2 (3)

1 - NOTIFY UDP Packet – new DLNA host is available



2 - Connect to host



3 - XML data containing the icon image

```
<iconList>
  <icon>
    <mimetype>image/png</mimetype>
    <width>48</width>
    <height>48</height>
    <depth>32</depth>

    <url>/images/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa [...]
```

Samsung #3

- **Date:** 2012
- **Tested device:** Samsung SmartTV D6000
- **Vulnerability:** Persistent Endless Loop
- **Details:** The controller packet contains a string, which is used to identify the controller itself. A malformed string will trigger an endless loop. The only way to fix this issue is to access the device **service mode** before the reboot.



`\n\n\n\t\t\t\tHACKED!\n\n\n\n\n`



2 - Allow/Deny controller



3 - Endless Loop



Philips Miracast (1)

- Found in 2014
- ALL the Philips TV 2013 models are affected
- Silently exploitable probably from Summer 2013
- No PIN
- No authorization request
- Hardcoded fixed password... “Miracast” 😊
- Full access to the TV services just like in LAN
- Exploiting of directory traversal in JointSpace
- Abuse of the available services
- Let’s check what we can do...

Philips Miracast (2)

- o Controlling the TV from remote



Philips Miracast (3)

- o Sending audio and video to the TV... the TV is talking to you!

The image is a composite of two parts. On the left, a video frame shows a camera on a tripod with the text "Why watching TV? We can PLAY on TV!" overlaid. Below this, the text "Transmit video remotely via DirectFB" is written in white on a black background. On the right, a Windows XP desktop is shown with a blue background. A file explorer window is open, displaying the contents of the "z:\directfb_test.exe" directory. Overlaid on the desktop is a DirectFB application window titled "z:\directfb_test.exe". The window's content area shows a log of system messages, including "Direct/Thread: Started 'Hoodoo 10' (-1) (DEFAULT PIPE-OTHER R-141 C...", "Usbdev/Message: Created remote super interface 'DirectFB'", and several "Direct/Interface: Using 'Requester' implementation of 'DirectFBDispla...", "Direct/Interface: Using 'Requester' implementation of 'DirectFBWindow...", "Direct/Interface: Using 'Requester' implementation of 'DirectFBSurfa...", "Direct/Interface: Using 'Dispatcher' implementation of 'DirectFBEvent...", "Direct/Interface: Using 'Dispatcher' implementation of 'DirectFBDataB...", and "Direct/Interface: Using 'Requester' implementation of 'DirectFBFont'...". The taskbar at the bottom shows the system clock as 11:45 AM on 3/18/2014.

Philips Miracast (4)

- Stealing configuration files and cookies via a directory traversal public from September 2013 but unfixed



What's next?

- o Android will be adopted on the upcoming SmartTV models:
 - o One platform makes exploit development easier
 - o Same vulnerable App will be used across different Vendors..
 - o Less customized software means less vulnerabilities ☹



Conclusion

- SmartTV are insecure
- SmartTV are a perfect target for “monitoring” a specific target: a person or even a company (TVs are everywhere)
- Exploiting them usually requires to be in the LAN or some user interaction
- Currently it's difficult to have a vulnerability for owning many models of TV, so you must focus on the one of your target



Thanks!

revuln.com
info@revuln.com
twitter.com/revuln