

RÉTRO-INGÉNIERIE

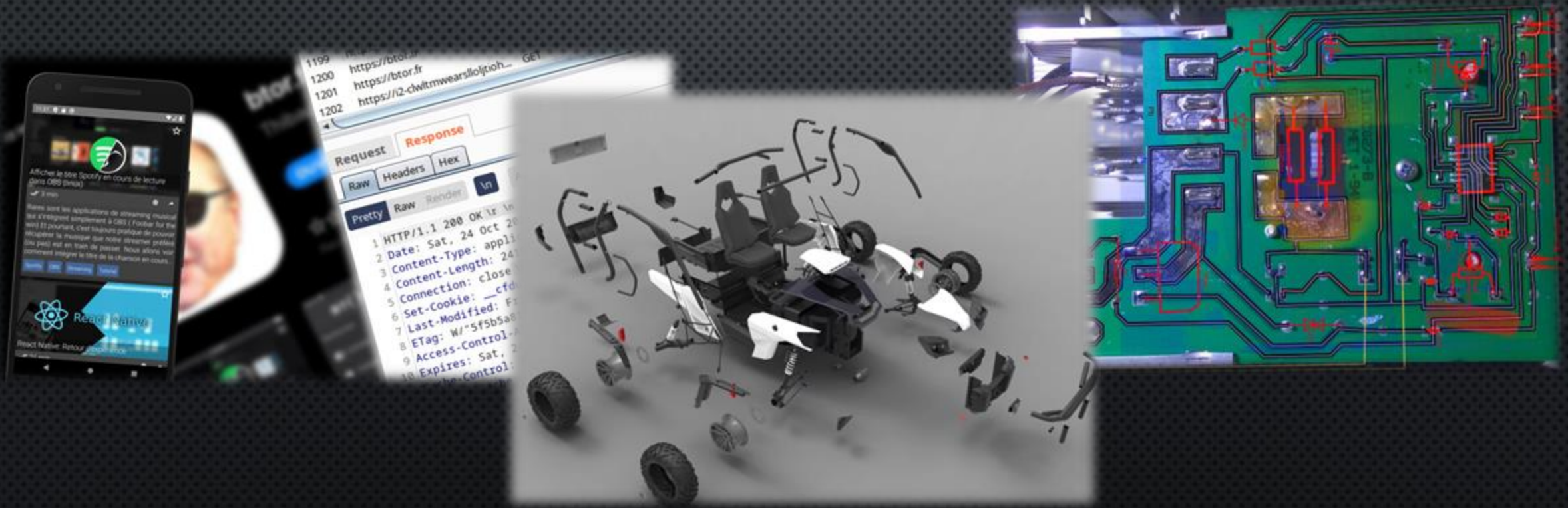


ANTHONY SEMAL

BECODE 2022

C'EST QUOI?

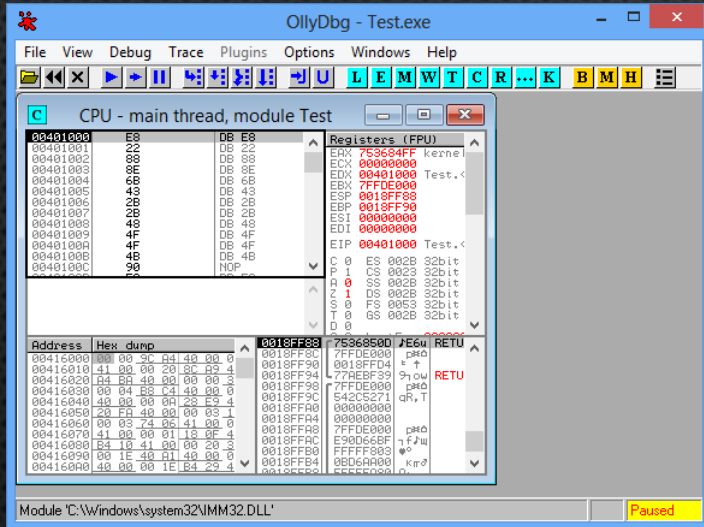
RÉALISER UN SCHÉMA, UN CODE, UNE DOCUMENTATION À PARTIR DE QUELQUE CHOSE QUI EXISTE (LOGICIEL OU MATERIEL)



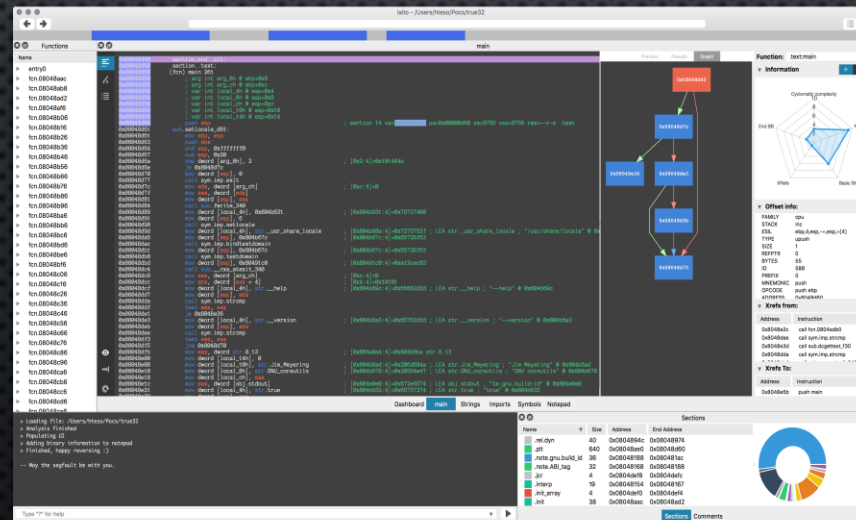
AU NIVEAU LOGICIEL

IL EXISTE 3 TYPES D'OUTILS POUR ANALYSE UN LOGICIEL

DÉBOGEUR



DÉSAMBLEUR



DÉCOMPILATEUR

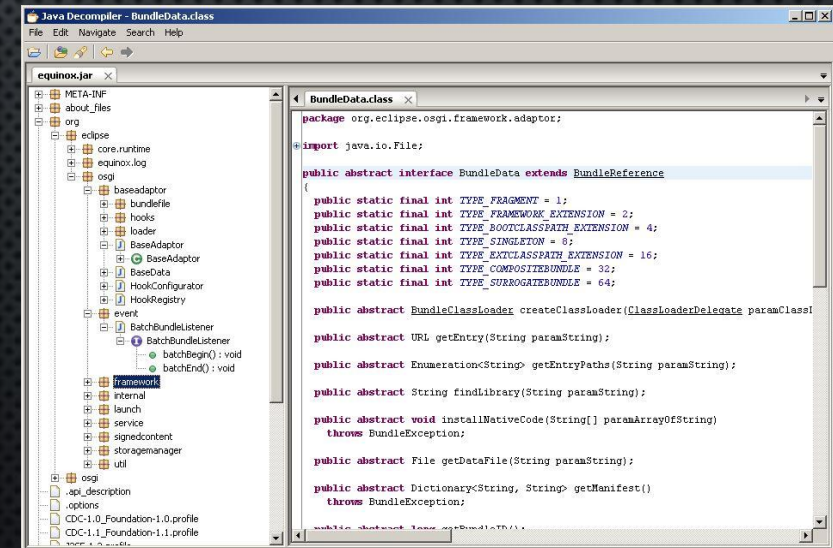
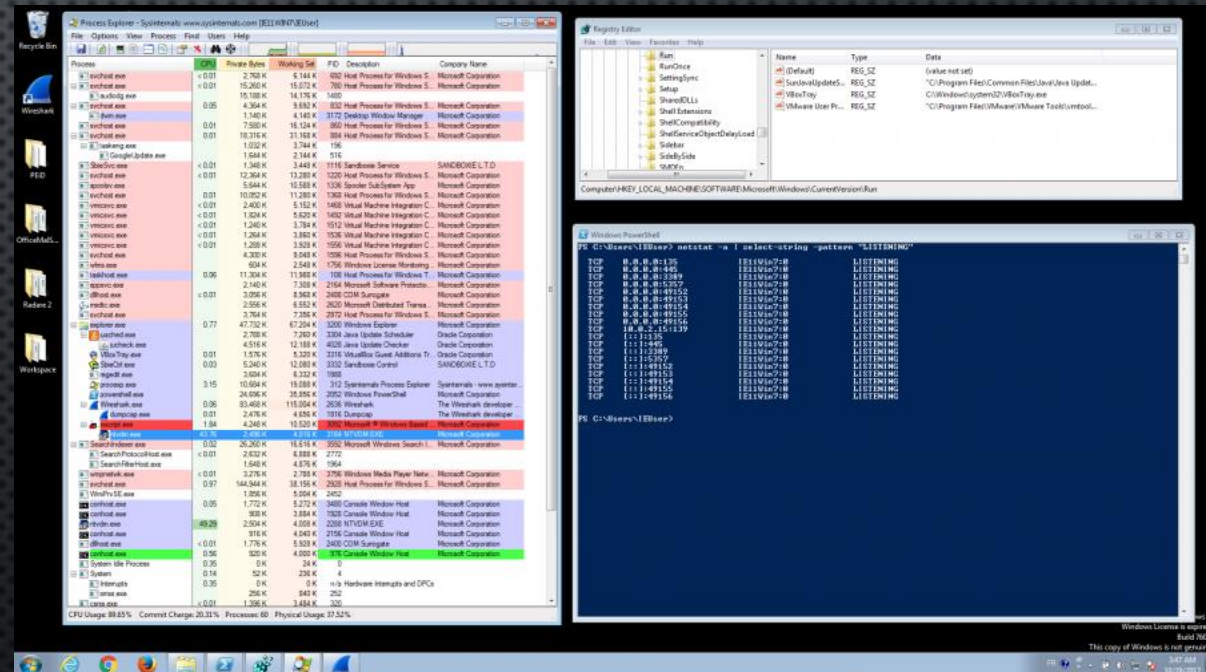


TABLEAU DES LOGICIELS DISPONIBLE (NON-EXHAUSTIVE)

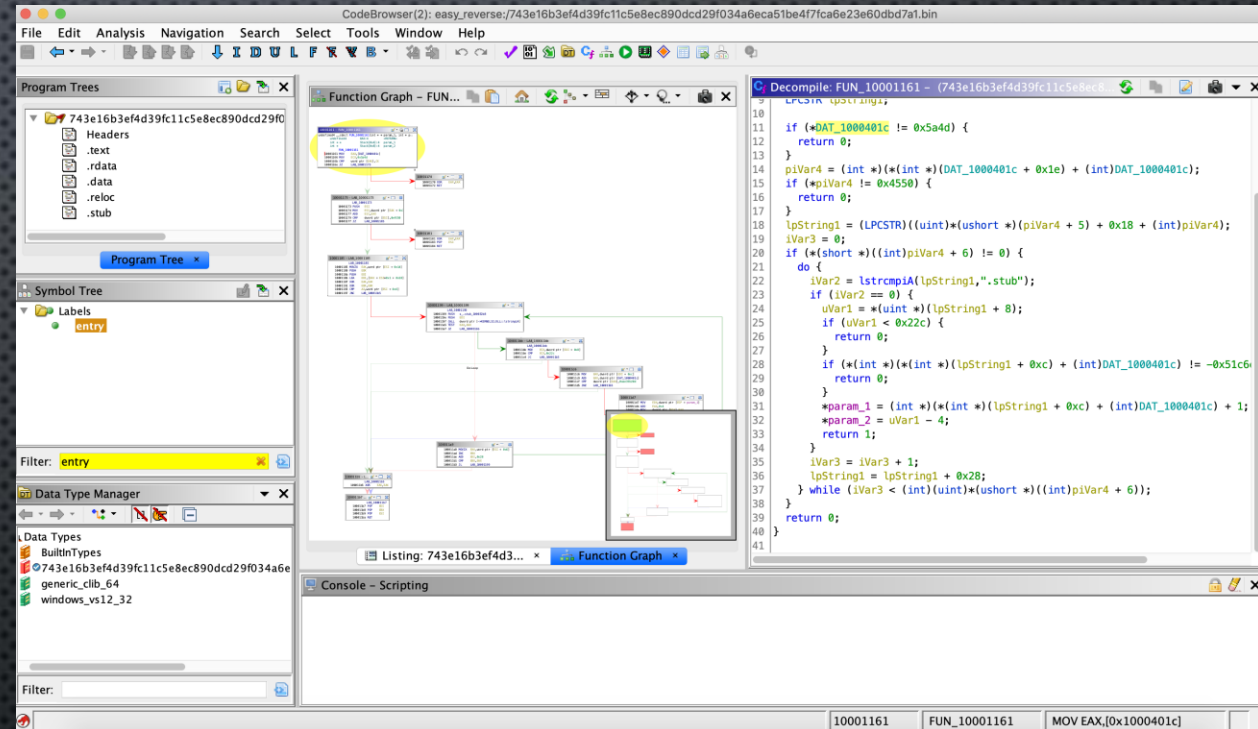
	AIO	DÉBOGUEUR	Désassembleur	Décompilateur
Windows	IDA (free/pro) Ghidra Ressource-hacker	Ollydbg X64dbg Windbg REDasm	Radare2 Ollydbg	Spice.decompilateur ILSpy DotPeek Java-Décompiler
Linux	IDA (free/pro) Ghidra	GDB (GNU Debugger) Valgrind	Radare2	Java-Decompiler Snowman C++ Decompiler
Android	IDA (free/pro) Ghidra	Android Debug Bridge	Dex2jar Apktool	Decompiler.com
MAC	IDA (free/pro) Ghidra		Radare2	
Editeur hexadécimale	HxD hex Editor MiniPro			

ANALYSE STATIQUE



ANALYSE STATIQUE

ON CHERCHE À COMPRENDRE LE
FONCTIONNEMENT D'UN MALWARE SANS
L'EXÉCUTER



ANALYSE DYNAMIQUE

CONSISTE À EXECUTER DIRECTEMENT LE
MALWARE DANS UN ENVIRONNEMENT
CONTROLÉ

QUELQUES OUTILS:

- LES OUTILS SYSINTERNALS.
- SYSCANALYZER: ANALYSEUR SYSTÈME AUTOMATIQUE DE MALWARE
- LORDPE:OUTIL QUI VA ANALYSER LA MÉMOIRE DE L'ORDINATEUR.
- WIRESHARK : ANALYSEUR RÉSEAU.

AU NIVEAU MATÉRIEL

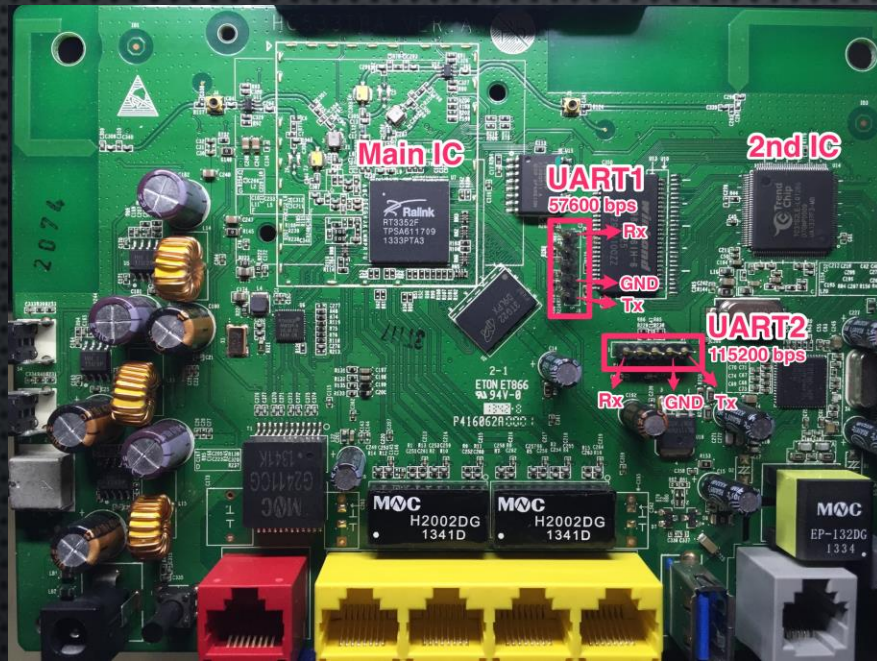
- RÉPARATION
- COPIER UN LOGICIEL PROPRIÉTAIRE
- TROUVER/EXPLOITER DES FAILLES (MELTDOWN/SPECTRE)

MAIS...

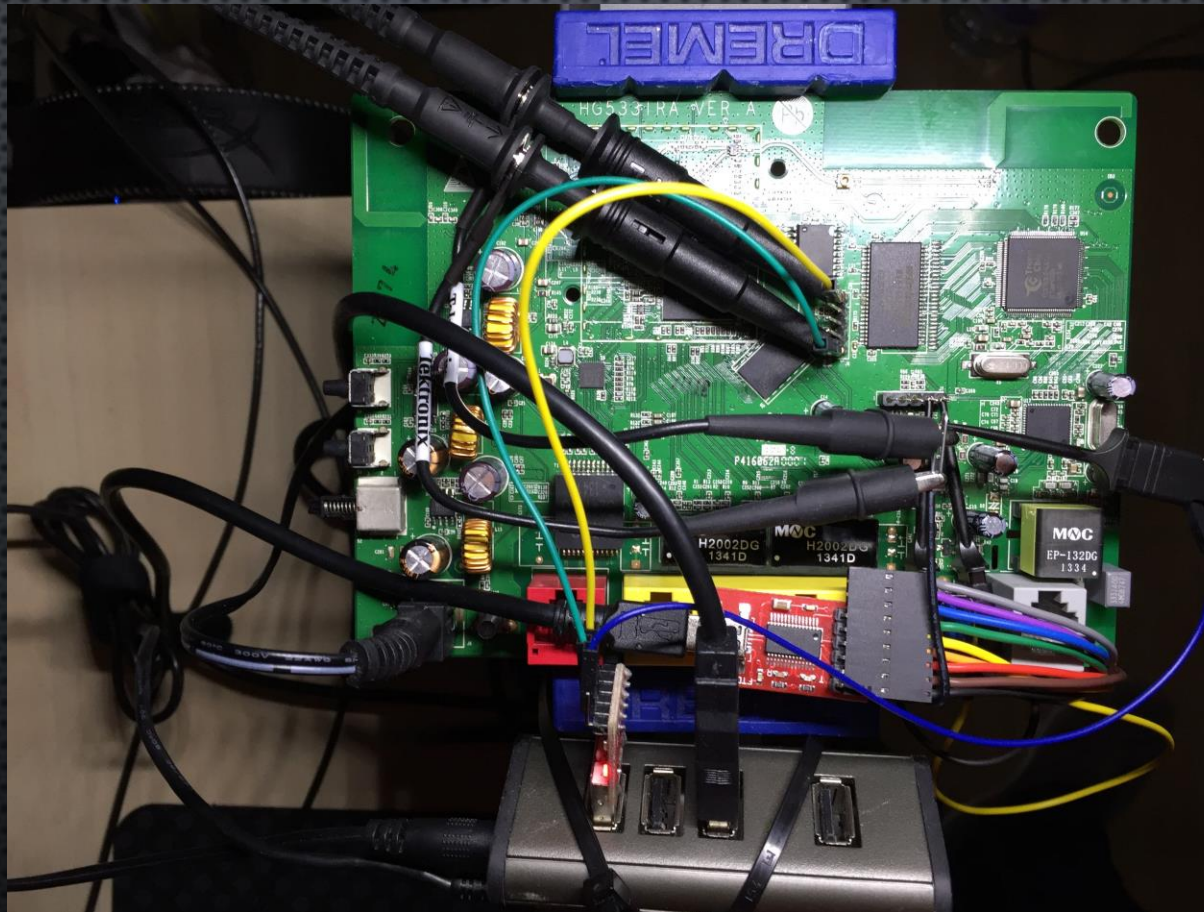
IL FAUT DU MATÉRIEL (PROGRAMMATEUR, OSCILLOSCOPE, ETC...

...ET S'Y CONNAITRE EN ÉLECTRONIQUE

ANALYSER LA CARTE ET IDENTIFIÉ LES COMPOSANTS RECHERCHÉ



TROUVER LES PORTS SÉRIES ET LES CONNECTER AU PC
(CONVERTISSEUR UART-USB)



OUVRIR UN TERMINALE ET "ECOUTER" LES PORTS SÉRIES

```
bash
~
$ miniterm.py /dev/tty.SLAB_USBtoUART 57600
--- Miniterm on /dev/tty.SLAB_USBtoUART: 57600,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---

U-Boot 1.1.3 (Aug 29 2013 - 11:16:19)

Board: Ralink APSoC DRAM: 64 MB
Reload Uboot WatchDog Timer.
spi device id: 1 2 16 4d 0 (2164d00)
find flash: $25FL064P
..RF_R17 = 0xA6
*** Warning - bad CRC, using default environment

Ralink UBoot Version: 3.5.2.0
ASIC 3352_MP (Port5<->None)
DRAM_TOTAL_WIDTH: 16 bits
TOTAL_MEMORY_SIZE: 64 Mbytes
Flash component: SPI Flash
Date:Aug 29 2013 Time:11:16:19
icache: sets:256, ways:4, linesz:32 ,total:32768
dcache: sets:128, ways:4, linesz:32 ,total:16384
.

netboot_common, argc= 2

KSEG1ADDR(NetTxPacket) = 0xA3FE4B00

NetLoop,call eth_halt !

NetLoop,call eth_init !
done

ETH_STATE_ACTIVE!!

Please choose operation:
  3: Boot system code via Flash (default).
  4: Entr boot command line interface.
  0
3: System Boot system code via Flash.
Reload Uboot WatchDog Timer.
## Booting image at bc020000 ...
. Image Name: HG533
  Created: 2013-08-29 3:16:32 UTC
  Image Type: MIPS Linux Kernel Image (lzma compressed)
  Data Size: 5648320 Bytes = 5.4 MB
  Load Address: 80000000
  Entry Point: 80390000
..... Verifying Checksum ... OK
Reload Uboot WatchDog Timer.

Python
~
$ miniterm.py /dev/tty.usbserial-AH02CEV2 115200
--- Miniterm on /dev/tty.usbserial-AH02CEV2: 115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---

Bootbase Version: VTC_SPI_BR1.1 | 2009/5/26 14:28:26
RAM: Size = 2048 Kbytes
DRAM POST: Testing: 2048K
OK
Found SPI Flash 512KiB W25X40 at 0xbfc00000

RAS Version: Huawei HG533 BT 20121127
System ID: *3.6.11.5(Y04.ZZ.3)3.12.8.220| 2012/02/27 | 2012/02/27

Press any key to enter debug mode within 3 seconds.
.....

Copyright (c) 2001 - 2006 TrendChip Technologies Corp.
TC2101MB_B13.2, initialize ch = 0, ethernet address: 00:aa:bb:01:23:45
Non Channel init ..... done
Initializing ADSL F/W ..... done
ANNEXAL
Press ENTER to continue...
ANNEXAL
SRADFF
Testlab 26
Dyingasp OFF!
Valid Loss of power OFF!
large0 flag=2 (0:maxD=64, 1:maxD=128, 2:maxD=511)
disable PMI
Sw patch status: OFF.
input line: tce
* phyaddr=1 Reg=26 value=9201
* phyaddr=1 Reg=00 value=2100
Auto Link OFF!
ok
```


TROUVER DES MOTS DE PASSE ROOT, INFORMATIONS SUR LA MÉMOIRE FLASH, ETC...

```
Mount-cache hash table entries: 512
bhal: bhalInit entry
deice id : 1 2 16 16 4d 0 (71641000)
S25FL064P (8192 Kbytes)
mtd .name = raspi, .size = 0x00800000 (8M) .erasize = 0x00010000 (64K) .numeraseregions = 0
27 05 19 56 de 1b c3 e0 52 1e bd 00 56 2f c0 80 00 00 00 80 39 00 00 c9 99 34 74 05 05 02 03 48 47 35 33 33 00 00
e2 bf 00 00 00 00 00 00 00 00 2e 4e 3d f6 03 00 00 00<5>Creating 4 MTD partitions on "raspi":
0x00000000-0x00020000 : "Bootloader"
0x00020000-0x0013d000 : "Main Kernel"
mtd: partition "Main Kernel" doesn't end on an erase block -- force read-only
0x0013d000-0x00660000 : "Main RootFS"
mtd: partition "Main RootFS" doesn't start on an erase block boundary -- force read-only
0x00660000-0x00800000 : "Protect"
NET: Registered protocol family 16
```

[illegible]

CONCLUSION

L'analyse de malware est un monde à part entière, vaste et complexe. Plus on apprend, et plus l'on comprend qu'on ne sait rien...

Valou

SOURCE

Introduction à la rétro-ingénierie

<https://dciets.com/dci-workshop/security/2020/01/29/Introduction-à-la-rétro-ingénierie/>

<https://clement-bouder.fr/Faire-de-la-retro-ingenierie-dans-le-cadre-dune-analyse-de-malware/>
avec une petite liste d'outils : <https://valou-tweak.fr/?p=2245>

Vidéo présentant l'analyse d'un keyword

<https://www.youtube.com/watch?v=NrFdBppIL0c>

Hardware

<https://korben.info/hardware-hacking.html>

<https://duo.com/blog/microcontroller-firmware-recovery-using-invasive-analysis>

<https://jcjc-dev.com/2016/12/14/reversing-huawei-5-reversing-firmware/>

<https://www.youtube.com/watch?v=vdEoQgTP0H4>

<https://connect.ed-diamond.com/misc/mischs-024/introduction-au-reverse-hardware>

https://airbus-seclab.github.io/hdd/SSTIC2015-Slides-hardware_re_for_software_reversers-czarny_rigo.pdf

<https://www.youtube.com/watch?v=vdEoQgTP0H4> - exploit contrôleur Bluetooth

<https://skyduino.wordpress.com/2013/11/10/tutohack-la-retro-ingenierie-cest-la-vie/>