

Copie de la mémoire et du disque via l'UEFI

Par Solal Jacob

Développeur de



digital forensics framework

DFF

Contenu

- La problématique
- BIOS et UEFI
- Bitlocker
- La solution

Problématique

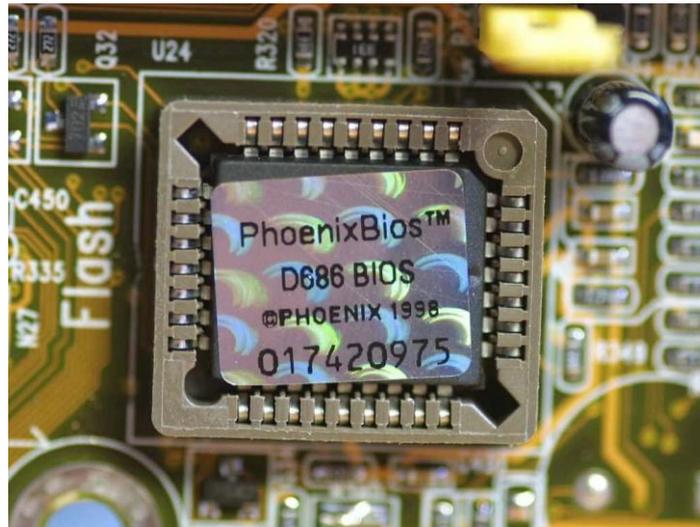


Mission d'analyse forensique d'un transformer book :

- Windows 10
- 1 port USB
- SSD soudé
- UEFI 32 bits

Le bios : principe de base

Basic Input Output System, créée en 1975 pour le système CP/M
Première instruction exécutée lorsque l'on appuie sur le bouton "on"



Le code :

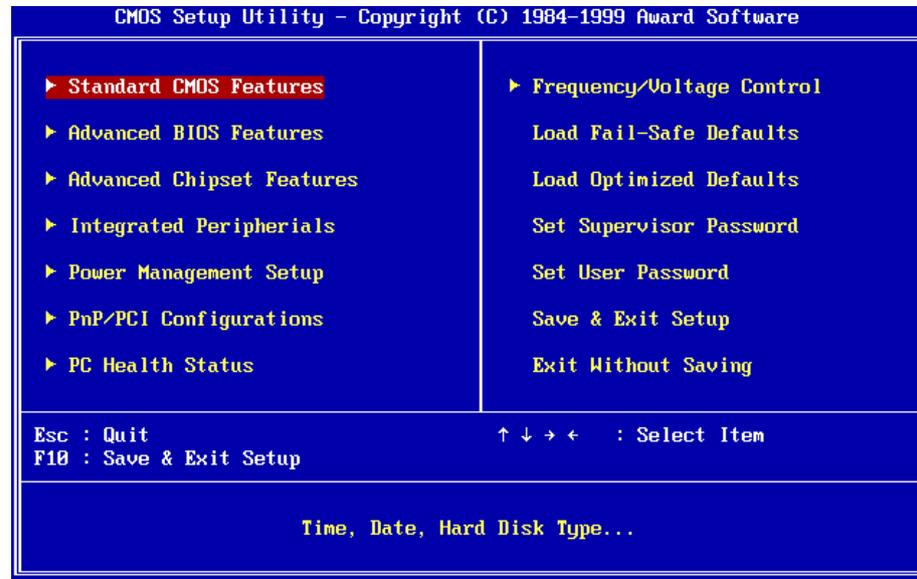
- Inscrit dans une mémoire non-volatile (EEPROM en général)
- Dépend du "chip-set" de la carte-mère
- Développé en ASM en général
- En source fermée (Projet coreboot pour un code ouvert)

Le bios : fonctionnement



- Un signal “reset” est envoyé par le “chipset” à la “CPU”
- La CPU démarre et lit ce qui se trouve à l’adresse 0xFFFF0 dans la ROM
- On y trouve une instruction “jump” qui pointe vers le début du code du BIOS
- L’affichage et le clavier sont activés
- Puis les autres périphériques (disque dur, cd-rom, carte réseau, bus USB, ...)
- L’interface graphique du BIOS devient accessible

Le bios : interface utilisateur



- Sélectionner un périphérique de démarrage
- Régler l'horloge interne
- Configurer le matériel
- Activer, désactiver les périphériques
- Configurer un mot de passe pour protéger le démarrage ou les paramètres
- Afficher certaines informations système sur les disques, la mémoire, ...

Le bios : la séquence de démarrage

Si le périphérique de démarrage est un disque dur :

La “zone d’amorçage” (Master Boot Record) est lue depuis le tout début du disque (premier cylindre, première piste, premier secteur)

- Elle contient le code de démarrage et la table des partitions

Ce code est exécuté et va démarrer :

- Directement le système d’exploitation
- Un “boot loader” (GRUB, LILO, MS)
qui va permettre de choisir le système à exécuter

La table des partitions

4 Partitions principales contenant :

Adresse CHS de début et de fin

Adresse LBA de début

Nombre de secteurs utilisés par la partition

Type de partition (FAT, NTFS, EXTFS)

Flags (bootable ou non)

1 ou des partitions étendues contenant:

Des partitions secondaires

Supporte une taille maximum de disque de 2 TO

UEFI le remplaçant du BIOS

1998 : Intel crée le standard
Extensible Firmware Interface

2006 : UEFI remplace maintenant EFI

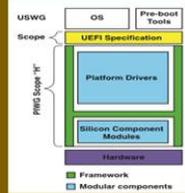


- Une interface logiciel entre le “Firmware” et le système d’exploitation
- Le “U” est apparu quand d’autres industriels se sont joints au standard (AMD, American Megatrends, Apple, Dell, HP, IBM, Insyde, Intel, Lenovo, Microsoft, and Phoenix Technologies)
- Peut tourner au-dessus ou a la place du BIOS
- Démarre le système plus rapidement

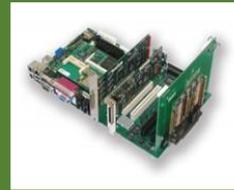
UEFI : Advantage



Mostly written in C.
High code re-use.



Emphasis on Specifications.
Standards compliance.



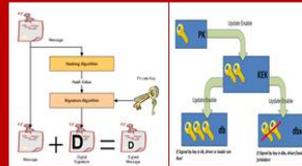
Better platform scaling. For e.g. removes shadow ROM limits.



Storage.
GPT removes 2.2 TB MBR restriction.



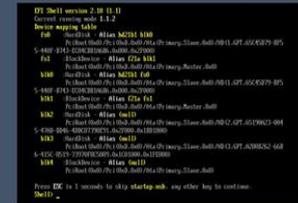
CPU Architecture independent. Platform design flexibility.



Secure boot solves "trust" related system integration challenges.



Pre-boot Networking.
Ipv4, Ipv6, PXE, VLAN, iSCSI etc.



UEFI shell improves pre-boot testing & diagnostics experience.

UEFI : Architecture

- SECurity : exécution des processus d'authentification et de contrôle d'intégrité (SecureBoot, mot de passe, token USB)
- Pre Efi Initialization : Initialisation de la carte-mère et du chipset
- Driver eXecution Environment : enregistrement de tous les pilotes
- Boot Device Selection : gestionnaire de démarrage comme grub
- Transient System Load : phase transitoire où le système d'exploitation est chargé
- RunTime : le système d'exploitation a pris la main
Interaction via des variables EFI stockées dans la NVRAM

UEFI : Shell

Développé par Intel

Ressemble à un shell Unix ou DOS

Permet de lancer des applications “UEFI”

Interpréteur python ou ruby

Possède de nombreuses commandes :

- Redirection
- Lister les partitions
- Monter un système de fichiers (fat)
- Info sur la RAM
- Info sur les périphériques
- Editeur hexadécimal (RAM & DISK)

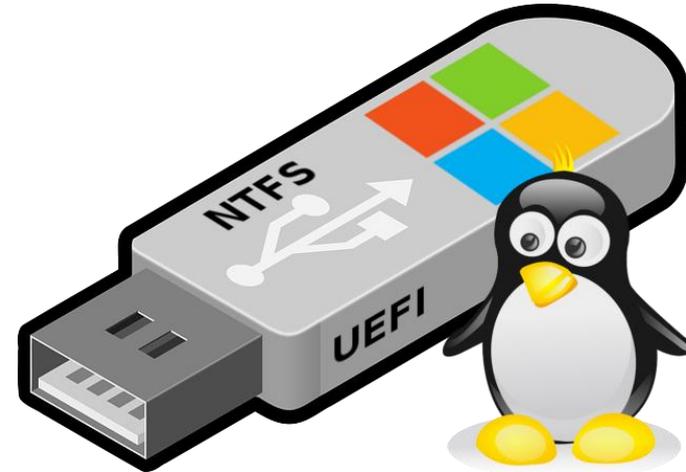
Contient déjà beaucoup de choses utiles pour une analyse forensique

Il peut être chargé via un clef usb ainsi que d'autres applications ...

```
EFI Shell version 2.00 [4096.1]
Current running mode 1.1.2
Device mapping table
fs0      :Removable HardDisk - Alias hd52g0b blk0
         Acpi (PNP0A03,0) /Pci (1D17) /Usb (6,0) /HD (Part1,Sig90909090)
blk0     :Removable HardDisk - Alias hd52g0b fs0
         Acpi (PNP0A03,0) /Pci (1D17) /Usb (6,0) /HD (Part1,Sig90909090)
blk1     :HardDisk - Alias (null)
         Acpi (PNP0A03,0) /Pci (1F12) /Ata (Primary,Master) /HD (Part1,SigD5BAE38B)
blk2     :HardDisk - Alias (null)
         Acpi (PNP0A03,0) /Pci (1F12) /Ata (Primary,Master) /HD (Part2,SigD5BAE38B)
blk3     :BlockDevice - Alias (null)
         Acpi (PNP0A03,0) /Pci (1F12) /Ata (Primary,Master)
blk4     :BlockDevice - Alias (null)
         Acpi (PNP0A03,0) /Pci (1F12) /Ata (Secondary,Master)
blk5     :Removable BlockDevice - Alias (null)
         Acpi (PNP0A03,0) /Pci (1D17) /Usb (6,0)

Press ESC in 1 seconds to skip startup.nsh, any other key to continue.
Shell> _
```

Analyse



Démarrage via Linux (Mint) en USB
Copie du disque :

« Freeze » de Linux (image partielle)

Problème : Chiffré avec Bitlocker

The screenshot shows a forensic tool interface with a file system tree on the left and a hex dump at the bottom. The file system tree shows a logical file named 'hdd_dump.0...hdd_dump.19' containing a partition table. The partition table lists several partitions, including Basic data partitions, an EFI system partition, a Microsoft reserved partition, and unallocated space. The hex dump at the bottom shows the raw data of a partition, with the first few lines containing the string 'X.-FVE-FS-.....' followed by '.....?.....' and '.....NO.NAME..'. The hex dump is displayed in hexadecimal and ASCII format.

name	size	tags	path
Basic data partition	734003200		/Logical/files/hdd_dump.0...hdd...
Basic data partition	5288025076		/Logical/files/hdd_dump.0...hdd...
Basic data partition	8588854562		/Logical/files/hdd_dump.0...hdd...
EFI system partition	104857600		/Logical/files/hdd_dump.0...hdd...
Microsoft reserved partition	134217728		/Logical/files/hdd_dump.0...hdd...
Unallocated	0		/Logical/files/hdd_dump.0...hdd...

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
00000000	58	90	2d	46	56	45	2d	46	53	2d	00	02	08	00	00	00	X.-FVE-FS-.....
00000010	00	00	00	00	f8	00	00	3f	00	ff	00	00	08	1d	00	00?.....
00000020	00	00	00	00	e0	1f	00	00	00	00	00	00	00	00	00	00
00000030	01	00	06	00	00	00	00	00	00	00	00	00	00	00	00	00
00000040	80	00	29	00	00	00	00	4e	4f	20	4e	41	4d	45	20	20	..).....NO.NAME..
00000050	20	20	46	41	54	33	32	20	20	20	33	c9	8e	d1	bc	f4	..FAT32...3.....
00000060	7b	8e	c1	8e	d9	bd	00	7c	a0	fb	7d	b4	7d	fb	f0	ac	(.....)

Bitlocker : Fonctionnement

Chiffrement complet du disque

Chiffrement basé sur AES

Stockage des clefs sur TPM, Clef USB, Carte à puce

Deux partitions :

- Partition de démarrage non chiffré

- Partition chiffrée (système de fichiers NTFS)

En-tête NTFS modifiée contenant des infos sur:

- La taille du système de fichiers

- Les métadonnées liées aux différentes clefs

- Clefs du disque chiffrées par la clef utilisateur



Bitlocker : Outils pour déchiffrer

LIBBDE : (Joachim Metz)

Formats supportés :

- Windows Vista
- Windows 7
- Windows 8 (Consumer Preview)
- BitLocker To Go

Méthodes de stockage des clefs supportées:

- Clear key
- Password
- Recovery password
- Start-up key
- FVEK (Full Volume Encryption Key) et TWEAK

Bitlocker : Outils pour déchiffrer

Dislocker : (Romain Coltel)

- Support de Windows 10 AES-XTS
- Permet de récupérer les clefs par défaut
- Information complète sur les métadonnées
- Support de l'écriture
- Code bien structuré et lisible

Bitlocker Attaque : Recherche des clefs en mémoire



Module volatility bitlocker

Nécessite une copie mémoire compatible avec volatility
Ou un fichier hibernation
Recherche via « pattern » spécifique autour de la clef
(Pool tag : FVEc, Cngb)

AES Key Finder :

Format RAW

Algorithme générique de détection de clefs AES

Copie de la RAM : Attaque « cold boot »



Lest We Remember : Cold Boot Attacks on Encryption Keys
(<http://citp.princeton.edu/memory>)

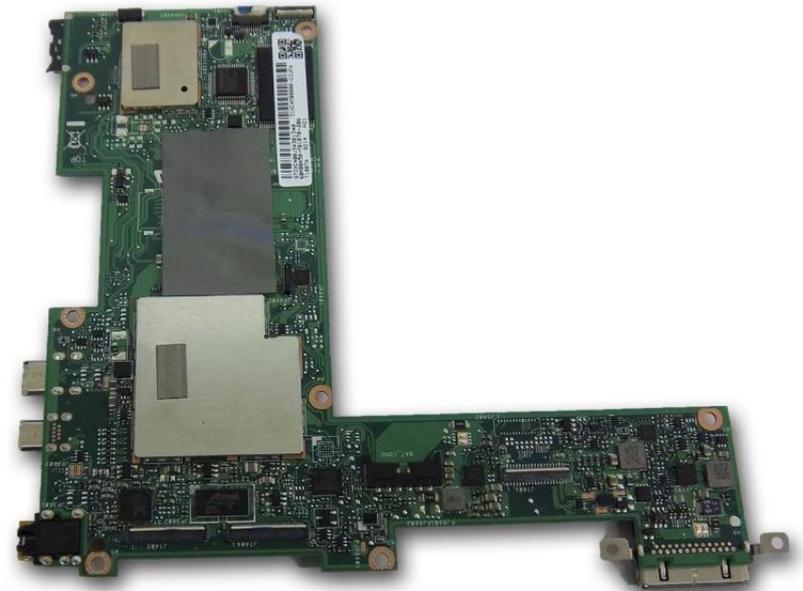
Problème :

- RAM soudée
- Code non maintenu
- Copie sur une clefs usb via BIOS/MBR
- Démarrage EFI copie via PXE

Problématique

RAM soudée

Pas de copie complète du « SSD »



Solution

Développement d'une application UEFI

```
EFI Shell version 2.00 [4096.1]
Current running mode 1.1.2
Device mapping table
fs0      :Removable HardDisk - Alias hd52g0b blk0
         Acpi (PNP0A03,0)/Pci (1D17)/Usb (6,0)/HD (Part1,Sig90909090)
blk0     :Removable HardDisk - Alias hd52g0b fs0
         Acpi (PNP0A03,0)/Pci (1D17)/Usb (6,0)/HD (Part1,Sig90909090)
blk1     :HardDisk - Alias (null)
         Acpi (PNP0A03,0)/Pci (1F12)/Ata (Primary,Master)/HD (Part1,SigD5BAE38B)
blk2     :HardDisk - Alias (null)
         Acpi (PNP0A03,0)/Pci (1F12)/Ata (Primary,Master)/HD (Part2,SigD5BAE38B)
blk3     :BlockDevice - Alias (null)
         Acpi (PNP0A03,0)/Pci (1F12)/Ata (Primary,Master)
blk4     :BlockDevice - Alias (null)
         Acpi (PNP0A03,0)/Pci (1F12)/Ata (Secondary,Master)
blk5     :Removable BlockDevice - Alias (null)
         Acpi (PNP0A03,0)/Pci (1D17)/Usb (6,0)

Press ESC in 1 seconds to skip startup.nsh, any other key to continue.
Shell> _
```

Copie le contenu de la RAM

Copie le contenu du disque

Les fichiers sont placés sur une clef ou un disque USB (FAT)

Découpés en fichiers de 4Go

Solution : outil de développement

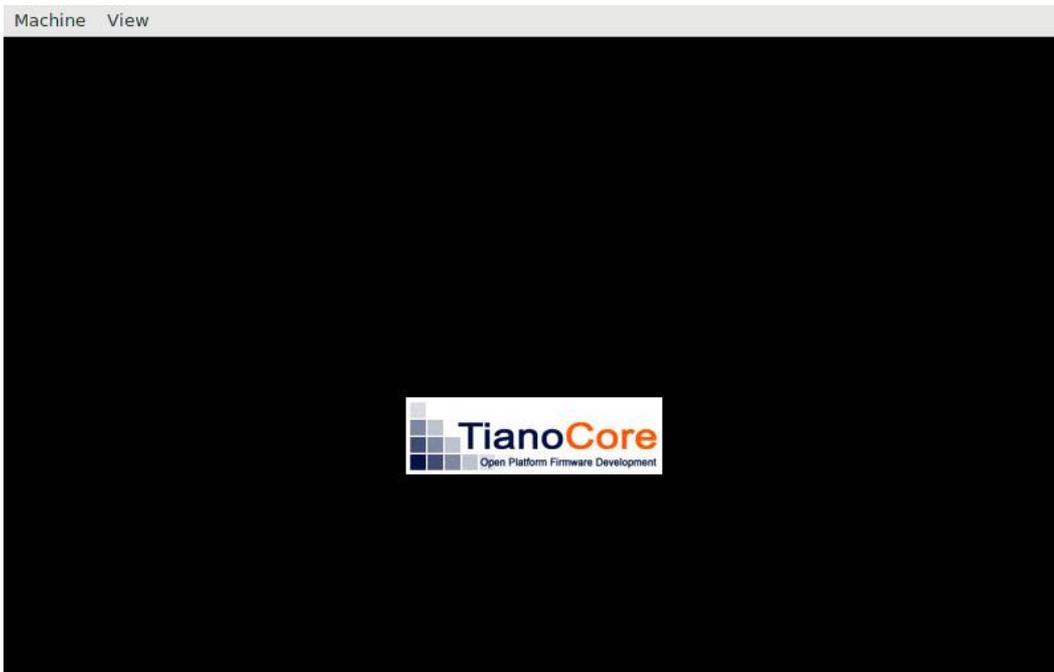
Utilisation du SDK Intel Edk2 TianoCore

<http://www.tianocore.org/edk2>

Développement en C

API complète

Génération d'exécutable (PE) 32 & 64 bits & drivers



UEFI pour QEMU :

Très lent au redémarrage

Pas de données en RAM

Résultat : Copie de la mémoire

```
fs3:\> Dumper.efi
ArxSys EFI RAM & Disk forensics dumper v0.1.
Available memory pages 465366 total size 1906139136
Creating memory dump in fs3:\memory.dump
Dumping range 0 - 8EFFF
Dumping range 90000 - 9DFFF
Dumping range 100000 - 1FFFFFFF
Dumping range 20100000 - 6F017FFF
Dumping range 6F019000 - 6F019FFF
Dumping range 76020000 - 78B2AFFF
Dumping range 79BAA000 - 79BAAFFF
Dumping range 79BB1000 - 79BC4FFF
Memory dumped ! (1906139136
bytes )
fs3:\> ls
Directory of: fs3:\

05/20/16  12:31p <DIR>          32,768  efi
05/19/16  05:15p                32,416  dmem.efi
05/20/16  03:12p                20,096  Dumper.efi
05/20/16  03:23p          1,906,139,136  memory.dump
          3 File(s) 1,906,191,648 bytes
          1 Dir(s)

fs3:\> _
```

Résultat : Copie du disque

```
Removable media 0 partition : size 1749642452480 number of blocks 3417270415
Media 0 block device: size 62545461248 number of blocks 122159104
Media 0 partition: size 104857600 number of blocks 204800
Media 0 partition: size 734003200 number of blocks 1433600
Media 0 partition: size 134217728 number of blocks 262144
Media 0 partition: size 52980350976 number of blocks 103477248
Media 0 partition: size 8589934592 number of blocks 16777216
Dumping media 0 of size 62545461248 to files hdd_dump
Creating new file fs4:\hdd_dump.0
Creating new file fs4:\hdd_dump.1
Creating new file fs4:\hdd_dump.2
Creating new file fs4:\hdd_dump.3
Creating new file fs4:\hdd_dump.4
Creating new file fs4:\hdd_dump.5
Creating new file fs4:\hdd_dump.6
Creating new file fs4:\hdd_dump.7
Creating new file fs4:\hdd_dump.8
Creating new file fs4:\hdd_dump.9
Creating new file fs4:\hdd_dump.10
Creating new file fs4:\hdd_dump.11
Creating new file fs4:\hdd_dump.12
Creating new file fs4:\hdd_dump.13
Creating new file fs4:\hdd_dump.14
Creating new file fs4:\hdd_dump.15
Creating new file fs4:\hdd_dump.16
Creating new file fs4:\hdd_dump.17
Creating new file fs4:\hdd_dump.18
Creating new file fs4:\hdd_dump.19
Disk dumped
fs4:\> _
```

Résultat : Extraction de la clef de chiffrement

```
MEMEX qemu # m^Cnt fat,img /mnt/loop
MEMEX qemu # mount /dev/sde1 /mnt/sde1/
MEMEX qemu # ls /mnt/sde1/
memory,efi Dumper,efi efi memory.dump
MEMEX qemu # ls /mnt/sde1/memory.dump ^C
MEMEX qemu # /home/vertrex/src/aeskeyfind/aeskeyfind /mnt/sd
sdb1/ sdb2/ sdc1/ sdc2/ sdc3/ sdd1/ sdd2/ sdd3/ sde1/ sde2/ sde3/
MEMEX qemu # /home/vertrex/src/aeskeyfind/aeskeyfind /mnt/sde1/memory.d
d7a768dcea595371e823424242424242
d7a768dcea595371e623424242424242
0477db071433187762424242424242
Keyfind progress: 100%
MEMEX qemu # □
```

Avantages

- Très simple d'utilisation ("Plug'n'play")
- Indépendant du système d'exploitation
- 32 & 64 Bits
- Ne nécessite pas d'identifiant, mot de passe
- Copie forensique
- Empreinte mémoire légère
- Ne nécessite pas de carte réseau
- Copie effectuée sur la clef ou le disque USB où se trouve l'exécutable
- Permet de récupérer des infos complémentaires pour l'analyse :
 - Sur les périphériques (Mémoire, disque, carte réseau, ...)
 - Heure système
 - ...

Améliorations futures

- Fournir un binaire précompilé
- Menu de sélection des différents choix
- Détection automatique des disques chiffrés
- Intégrer la sortie directement dans un logiciel forensique (ex: DFF)
- Recherche de la clef et déchiffrement automatique des disques

Questions

