

Récupération de données

Manuel vérifié et sourcé — Édition mai 2026

Anatomie des supports, méthodes professionnelles, outils ouverts et commerciaux, prévention.

Toutes les affirmations chiffrées sont sourcées ou explicitement qualifiées d'ordres de grandeur. Aucun cas n'est inventé : les incidents cités sont publics et référencés.

Édition révisée

Document de référence pour techniciens, étudiants en forensique et particuliers avancés.

Avertissement et méthode

Ce livre est issu de la révision critique d'un corpus pédagogique initial qui présentait de nombreuses inventions chiffrées sous le vernis de « sources professionnelles ». Le présent texte remet le travail sur pied en respectant trois règles :

- Chaque pourcentage est sourcé ou explicitement étiqueté comme ordre de grandeur quand aucune statistique publique ne le justifie.
- Aucun cas n'est inventé. Les études de cas sont des incidents publics référencés (NotPetya/Maersk 2017, Code Spaces 2014, etc.).
- Les versions de logiciels et numéros propriétaires ne sont mentionnés que lorsqu'ils sont vérifiables auprès de l'éditeur.

À qui s'adresse ce livre

Aux techniciens IT qui doivent gérer des incidents de perte de données ; aux étudiants en forensique numérique qui cherchent une vue d'ensemble pédagogique ; aux particuliers avancés qui veulent comprendre ce qui se passe sous le capot avant d'agir ou de confier leur support à un laboratoire ; aux juristes et experts judiciaires qui ont besoin d'une vue technique pour évaluer la recevabilité d'éléments numériques.

À qui il ne s'adresse pas

À quelqu'un qui a une urgence active et n'a pas le temps de lire. Dans ce cas, la règle est simple : **débrancher le support, ne rien y écrire, et soit consulter le chapitre 5 (diagnostic) pour se positionner, soit contacter directement un laboratoire.** Ce livre est un manuel de réflexion, pas un guide de survie.

Avertissement légal

La récupération de données sur des supports qui ne vous appartiennent pas, ou qui contiennent des données protégées par le secret professionnel ou le RGPD, est encadrée par la loi. Les techniques décrites ici sont à usage éducatif et professionnel uniquement. Pour tout enjeu sérieux — données médicales, judiciaires, propriété intellectuelle — passer par un laboratoire sous contrat de confidentialité reste la seule voie acceptable.

Sommaire

Introduction

La récupération de données en 2026

Partie I — Fondations physiques

Chapitre 1 — Anatomie d'un disque dur mécanique (HDD)

Chapitre 2 — Anatomie d'un SSD : NAND, contrôleur, FTL

Chapitre 3 — Systèmes de fichiers : la carte du trésor

Partie II — Diagnostic

Chapitre 4 — Causes de perte : les chiffres 2025-2026

Chapitre 5 — Diagnostic et triage

Partie III — Méthodes

Chapitre 6 — Imagerie sécurisée : la fondation de tout

Chapitre 7 — Analyse logique et réparation FS

Chapitre 8 — Data carving en profondeur

Chapitre 9 — Intervention physique sur HDD

Chapitre 10 — Intervention physique sur SSD

Chapitre 11 — RAID et stockage avancé

Partie IV — Cas spéciaux

Chapitre 12 — Chiffrement et récupération

Chapitre 13 — Supports mobiles (Android, iOS)

Chapitre 14 — Forensique judiciaire

Partie V — Pratique

Chapitre 15 — Outils 2026 : panorama réaliste

Chapitre 16 — Pièges mortels et études de cas

Partie VI — Prévention

Chapitre 17 — Stratégies de sauvegarde modernes

Chapitre 18 — Limites actuelles en 2026

Annexes

A. Commandes de référence

B. Glossaire

C. Bibliographie

INTRODUCTION

Chapitre 0

La récupération de données en 2026

Une discipline à l'intersection de plusieurs métiers

La récupération de données est l'ensemble des techniques permettant de retrouver des informations rendues inaccessibles sur un support de stockage. La discipline mobilise quatre familles de compétences : physique des matériaux (magnétisme pour les disques durs, électronique à grille flottante pour les mémoires NAND), algorithmique des systèmes de fichiers, ingénierie inverse de contrôleurs propriétaires, et rigueur procédurale forensique.

On la distingue soigneusement de la **restauration de sauvegarde**, qui relève de la prévention. La récupération intervient après la perte, sur un support qui n'a pas de copie utilisable. C'est, par construction, une discipline d'urgence où l'on fait avec ce qu'il reste.

Deux mondes, jamais à confondre

Récupération logique : le support est physiquement intact et détecté par la machine. Le problème est dans le logiciel — système de fichiers corrompu, partition supprimée, fichiers effacés, chiffrement par ransomware. Les données brutes sont presque toujours encore là ; il faut juste savoir les lire.

Récupération physique : panne matérielle. Le support n'est plus détecté, ou émet des bruits anormaux, ou son contrôleur ne répond plus. L'intervention demande un environnement spécialisé : salle blanche pour les disques durs mécaniques, station de micro-soudure pour les SSD.

Tout le travail de récupération commence par un diagnostic qui tranche entre ces deux mondes. Se tromper de monde coûte des données. Le chapitre 5 détaille cette étape.

L'évolution depuis 1990

Pendant trois décennies, la récupération s'est faite sur disques durs magnétiques. Le principe était stable : tant que les plateaux n'étaient pas physiquement réécrits, les données restaient en place. Suppression, formatage rapide, corruption logique : tout cela n'affectait que la table d'allocation, pas le contenu. C'était l'âge d'or des logiciels grand public type Norton Utilities ou, plus tard, TestDisk.

L'arrivée massive des SSD à partir de 2010 a fait basculer la discipline. La mémoire NAND ne se comporte pas comme un disque magnétique : on ne peut pas réécrire en place, et le contrôleur doit en permanence consolider et effacer des blocs entiers. Avec la commande **TRIM** (apparue dans Windows 7, macOS 10.6.8 et le noyau Linux 2.6.33), supprimer un fichier déclenche un effacement *physique* des cellules concernées, souvent en quelques secondes à minutes. Sur SSD moderne en bonne santé, la fenêtre de récupération logique se réduit dramatiquement.

Parallèlement, deux autres évolutions ont remodelé le paysage : la généralisation du **chiffrement matériel** (SED, TCG Opal, BitLocker avec TPM, FileVault) qui rend la donnée physique inutilisable sans la clé, et l'explosion des attaques par **ransomware** qui ciblent désormais explicitement les sauvegardes. Le Verizon DBIR 2025 chiffre cette dernière tendance : le ransomware est impliqué dans 44 % des compromissions de données documentées en 2024, en hausse de 37 % sur un an.

Sur les statistiques — Aucun statistique consolidée publique ne donne le taux de réussite global de la récupération de données. Les laboratoires professionnels publient parfois des chiffres dans leurs supports commerciaux, mais ce sont des chiffres non vérifiables et non indépendants. Ce manuel parle d'ordres de grandeur quand un chiffre précis serait malhonnête.

Comment ce livre est organisé

Six parties, dix-huit chapitres et trois annexes. La progression est délibérée : on commence par comprendre ce qu'est physiquement un support de stockage (partie I), puis comment diagnostiquer une panne (partie II), puis quelles méthodes appliquer (partie III), puis comment gérer les cas particuliers (partie IV), puis quels outils choisir dans la pratique (partie V), et enfin comment rendre tout cela inutile par une bonne prévention (partie VI).

Trois conventions de mise en page :

- Les encadrés **bleus** sont des notes pédagogiques.
- Les encadrés **orange** sont des avertissements opérationnels — à lire avant d'agir.
- Les encadrés **verts** sont des études de cas publiques et sourcées.

Partie I

Fondations physiques

Avant toute méthode, il faut comprendre *physiquement* comment une donnée est écrite, lue et supprimée. Trois chapitres : un disque dur mécanique (HDD), une mémoire NAND (SSD, eMMC, UFS, carte SD), un système de fichiers (la couche logique qui organise les blocs en arborescence). Sans cette base, les méthodes des chapitres suivants ne sont que de la magie.

PARTIE I — FONDATIONS PHYSIQUES

Chapitre 1

Anatomie d'un disque dur mécanique

1.1 Vue d'ensemble

Un disque dur (HDD, *Hard Disk Drive*) est une mécanique de précision miniaturisée. Sous son boîtier hermétique, on trouve : un ou plusieurs **plateaux** rigides revêtus d'une couche magnétique, un **moteur spindle** qui les fait tourner à vitesse constante, des **têtes de lecture/écriture** portées par un **bras actionneur** qui balaye la surface, et un **circuit imprimé** (PCB) extérieur qui contient le contrôleur, la ROM de firmware et l'interface SATA ou SAS.

Les ordres de grandeur typiques d'un HDD 3,5" grand public en 2026 :

Paramètre	Valeur typique
Vitesse de rotation	5 400 ou 7 200 tr/min (consumer), jusqu'à 15 000 (entreprise SAS)
Capacité par plateau	2 à 4 To
Nombre de plateaux	1 à 10 selon la capacité totale
Densité linéaire	> 1 million de bits par pouce de piste
Vol des têtes (fly height)	Quelques nanomètres au-dessus du plateau
Débit séquentiel	150 à 300 Mo/s
Latence d'accès aléatoire	5 à 15 ms

1.2 Comment une donnée est écrite

La couche magnétique du plateau est divisée en milliards de petits domaines magnétiques. La tête d'écriture, en générant un champ magnétique local très fort et très bref, oriente la polarisation d'un domaine dans un sens (bit 1) ou dans l'autre (bit 0). La transition entre deux polarisations opposées est ce que la tête de lecture détecte ensuite, par induction ou plus récemment par effet tunnel magnétorésistif (TMR).

Tant que rien ne réécrit la zone, ces polarisations magnétiques sont **stables sur des décennies**. C'est la propriété fondamentale qui rend les HDD si récupérables : effacer un fichier au niveau du système de fichiers ne touche pas aux domaines magnétiques eux-mêmes.

1.3 PMR, CMR, SMR, HAMR : les technologies d'écriture

Quatre technologies coexistent ou se succèdent :

- **PMR** (Perpendicular Magnetic Recording) : depuis 2005, les domaines magnétiques sont orientés perpendiculairement à la surface plutôt que parallèlement. C'est la base de tous les HDD modernes.
- **CMR** (Conventional Magnetic Recording) : terme moderne qui désigne le PMR « classique » avec des pistes écrites côte à côte sans recouvrement. Permet de réécrire n'importe quelle piste sans toucher aux voisines.

- **SMR** (Shingled Magnetic Recording) : depuis 2013, les pistes se recouvrent partiellement comme des tuiles de toit. On gagne 20 à 25 % de densité, mais chaque modification d'une piste oblige à réécrire toute la bande de pistes adjacentes. Le firmware doit gérer une zone de cache et un garbage collection comparable à celui d'un SSD.
- **HAMR** (Heat-Assisted Magnetic Recording) : technologie qui chauffe ponctuellement le domaine par laser pendant l'écriture pour réduire la taille des domaines stables. Commercialisée à partir de 2024 sur les disques entreprise très haute capacité (30 To+).

SMR et récupération — Le SMR est sensiblement plus complexe à récupérer en cas de panne firmware : la translation entre adresses logiques (LBA) et emplacements physiques sur la bande est gérée par le contrôleur, et un firmware corrompu peut rendre le contenu illisible même sur des plateaux intacts. Les laboratoires utilisent des modules spécialisés (PC-3000 a publié des modules SMR à partir de 2020) qui doivent gérer à la fois le translator principal et le cache translator. Source : Rossmann Group, *CMR vs SMR: How Recording Technology Affects Recovery*, 2026.

1.4 Causes typiques de panne mécanique

1. **Head crash.** Une tête entre en contact avec la surface du plateau — choc, vibration, défaut de fly height. Le résultat est souvent une rayure progressive qui détruit physiquement la couche magnétique. Symptôme classique : *clics répétés* de la tête qui cherche en vain à se positionner.
2. **Stiction.** Les têtes restent collées au plateau au lieu de se parquer correctement à l'arrêt. Le moteur n'arrive plus à lancer la rotation. Symptôme : *bruit de bourdonnement court puis silence*.
3. **Moteur spindle HS.** Les paliers s'usent, le moteur grippe. Symptôme : *plateau qui ne tourne plus du tout, ou tourne par à-coups*.
4. **PCB grillé.** Une surtension détruit des composants du circuit imprimé, souvent le TVS (transistor de protection). Le disque n'est plus détecté du tout, parfois il fume littéralement.
5. **Corruption firmware.** La ROM du PCB ou une zone système des plateaux (zone de service) devient illisible. Le disque tourne mais ne se monte pas, ou se monte avec une capacité absurde.

Attention — Un HDD qui claque doit être éteint immédiatement. Chaque rotation supplémentaire des plateaux quand les têtes touchent la surface étend la zone rayée — chaque seconde, on perd des données. C'est l'une des rares vraies urgences en récupération.

1.5 Pourquoi un HDD reste très récupérable

Quand un système de fichiers supprime un fichier, il modifie seulement ses propres tables internes (MFT pour NTFS, inodes pour ext4, table FAT pour FAT32/exFAT). Les secteurs physiques qui contenaient le fichier ne sont ni effacés, ni démagnétisés. Ils le seront seulement quand un nouveau fichier viendra écrire par-dessus.

C'est pour cela que sur un HDD :

- Un fichier supprimé est récupérable tant que ses secteurs n'ont pas été réutilisés.
- Un formatage rapide ne fait que réinitialiser les structures de base du FS ; les secteurs restent intacts.
- Un formatage complet (qui réécrit tout) détruit effectivement les données, mais prend des heures et n'est presque jamais fait par accident.

Sur un HDD non écrasé, les outils de récupération logique (TestDisk, R-Studio, UFS Explorer, PhotoRec en dernier recours) retrouvent en pratique l'écrasante majorité des données.

1.6 La fiabilité moyenne d'un HDD en 2026

Backblaze, hébergeur cloud, publie depuis 2013 les statistiques de panne de son parc de disques durs. Le rapport annuel 2025 (publié en février 2026) compte 344 196 disques répartis sur 30 modèles. Trois chiffres clés :

- AFR annuel 2025 : **1,36 %** (en baisse par rapport à 1,55 % en 2024).
- AFR sur la vie complète des disques (lifetime) : **1,30 %**, stable d'un trimestre à l'autre.
- Q4 2025 a affiché un AFR trimestriel de 1,13 %, le plus bas depuis 2022.

Autrement dit : sur un échantillon massif, environ 1,4 % des disques tombent en panne chaque année. Pour un particulier qui a un seul disque, cela ne dit pas grand-chose individuellement — votre disque personnel tombera en panne ou non, c'est binaire. Mais cela rappelle que sur un parc, la panne est statistiquement certaine.

Source : Backblaze, Drive Stats for 2025, rapport annuel publié le 12 février 2026 ; communiqué officiel BusinessWire.

PARTIE I — FONDATIONS PHYSIQUES**Chapitre 2****Anatomie d'un SSD : NAND, contrôleur, FTL****2.1 Un changement de paradigme**

Un SSD (*Solid State Drive*) n'a pas de pièces mécaniques. Toute la complexité est dans l'électronique. Cela paraît une simplification, mais pour la récupération, c'est l'inverse : la mémoire NAND impose des contraintes physiques qui forcent le contrôleur à effacer activement les données supprimées. Sur un HDD, supprimer ne détruit rien ; sur un SSD moderne, supprimer *détruit* dans les minutes qui suivent.

2.2 Les cellules NAND

L'unité fondamentale est la cellule NAND : un transistor à **grille flottante** (floating-gate MOSFET) ou, plus récemment, à piégeage de charge (charge-trap flash). Une charge électrique piégée dans la grille flottante modifie la tension de seuil du transistor. En mesurant cette tension, on lit la valeur stockée.

Selon le nombre de niveaux de tension qu'on distingue dans une même cellule, on stocke plus ou moins de bits. C'est le cœur du trade-off densité / endurance qui structure tout le marché du stockage flash :

Type	Bits/cellule	Niveaux	Endurance (cycles P/E)	Usage
SLC	1	2	50 000 à 100 000	Industriel, entreprise critique
MLC	2	4	3 000 à 10 000	Enterprise (en voie de disparition)
TLC	3	8	1 000 à 3 000	Consumer SSD courant 2026
QLC	4	16	150 à 1 000	Grande capacité bon marché
PLC	5	32	< 150 (estimation)	En développement, peu commercialisé

Sources : Kingston Technology, Lexar Enterprise, TechTarget, OSCOO. Les ordres de grandeur d'endurance varient selon les sources et les modèles précis ; ces valeurs représentent le haut de la fourchette typique en 2024-2026.

Plus on stocke de bits par cellule, plus la marge entre niveaux est étroite, plus les erreurs de lecture sont fréquentes, plus le contrôleur doit appliquer de codes correcteurs (ECC), et plus la cellule s'use vite à chaque cycle d'écriture/effacement.

3D NAND — Les SSD modernes utilisent presque tous une architecture 3D NAND, où les cellules sont empilées verticalement (couramment 96, 176, 232 couches en 2024-2026). Cela permet de revenir à des lithographies plus larges (donc à des cellules plus robustes) tout en augmentant la densité totale. Un SSD TLC 3D 2026 a souvent une endurance pratique meilleure qu'un MLC 2D des années 2010, malgré ses 3 bits par cellule.

2.3 La contrainte fondamentale : écrire à la page, effacer au bloc

C'est le détail qui explique tout le reste. La mémoire NAND peut être lue et écrite au niveau de la **page** (typiquement 4, 8 ou 16 Ko), mais elle ne peut être effacée qu'au niveau du **bloc** (256 à 512 pages, soit typiquement 1 à 8 Mo). On ne peut pas réécrire en place : il faut effacer le bloc entier d'abord.

Pour rester performant, le contrôleur ne fait jamais cette opération de manière naïve. Quand on modifie un fichier, il :

1. Écrit la nouvelle version dans une page libre, ailleurs sur la NAND.
2. Met à jour sa table interne de correspondance (mapping) pour que le LBA logique pointe désormais vers la nouvelle page.
3. Marque l'ancienne page comme « invalide » mais ne l'efface pas immédiatement.
4. Plus tard, en arrière-plan, le **garbage collector** consolide les pages valides restantes des blocs partiellement utilisés et applique la tension d'effacement sur les blocs vides.

2.4 Wear leveling et over-provisioning

Comme chaque cellule a une endurance limitée, le contrôleur applique du **wear leveling** : il répartit les écritures sur toutes les cellules disponibles, pour qu'aucune ne s'use plus vite que les autres. Une zone d'**over-provisioning** (7 % minimum, souvent 14 % à 28 % sur les SSD entreprise) est invisible pour l'utilisateur mais utilisable par le contrôleur pour ses opérations de réorganisation et pour remplacer les cellules qui finissent par mourir.

2.5 TRIM : la commande qui change tout

Sans TRIM, le contrôleur n'a aucune idée des pages qui sont encore utilisées au niveau du système de fichiers. Quand l'OS supprime un fichier, il modifie ses propres tables mais ne le dit pas au SSD. Résultat : le contrôleur considère ces pages comme contenant des données valides, et son garbage collector perd un temps fou à les déplacer inutilement.

TRIM (commande ATA DATA SET MANAGEMENT avec l'attribut Trim, ou son équivalent NVMe DEALLOCATE) résout ce problème en informant le contrôleur des LBA libres côté OS. Le contrôleur peut alors :

- Mettre à jour sa table de mapping immédiatement.
- Programmer ces blocs pour effacement physique au prochain cycle de garbage collection.

Sur la plupart des SSD modernes implémentant DRAT (*Deterministic Read After Trim*) ou DZAT (*Deterministic Zero After Trim*), toute lecture ultérieure des LBA trimés renvoie respectivement une valeur déterministe non spécifiée ou des zéros. Les outils de récupération logique ne voient plus rien d'utile.

2.6 La fenêtre de récupération

Combien de temps avant que les données soient physiquement perdues ? La référence sur le sujet — Rossmann Group — est claire : « *Most modern SSDs process TRIM within seconds to minutes, making any recovery window negligible* ».

Concrètement :

- Sur la suppression d'un fichier sur un SSD interne monté en NTFS/APFS/ext4 sous Windows/macOS/Linux récent : TRIM est envoyé immédiatement. La fenêtre se compte en secondes.
- Le garbage collector peut s'activer dès la prochaine période d'inactivité, donc en quelques minutes au maximum.
- Une fois le bloc effacé physiquement, aucun chip-off ne récupère quoi que ce soit. Les cellules sont à leur état neutre.

Attention — Pratiquement : si vous avez supprimé un fichier important sur un SSD, débranchez le support immédiatement, ne le rebranchez plus sur la même machine, et envoyez-le pour analyse. Chaque seconde sous tension réduit les chances. Et même en agissant vite, considérez que la perte est probable — pas certaine, mais probable.

2.7 Quand TRIM ne fonctionne pas

La récupération SSD reste possible dans plusieurs configurations où TRIM est court-circuité, documentées notamment par Belkasoft (*Recovering Evidence from SSD Drives*, Forensic Focus). Voici les principales :

- **RAID matériel.** La plupart des contrôleurs RAID ne passent pas TRIM aux disques sous-jacents.
- **SSD externes via vieux ponts USB-SATA.** Les JMicron JMS539 et ASMedia ASM1051 anciens ne passent pas TRIM. Les ponts UASP récents (JMS578, ASM235CM, RTL9210B) le passent.
- **NAS.** Selon le firmware et la configuration des volumes, TRIM peut être absent ou différé.
- **Pseudo-SSD bas de gamme.** Certaines clés USB et cartes SD marketées comme SSD n'implémentent pas TRIM.
- **Firmware buggé.** Plusieurs modèles (notamment des Crucial M4, OCZ Vertex, Intel 320) ont eu en sortie d'usine un TRIM cassé. Si l'utilisateur n'a jamais mis à jour, les données peuvent rester.
- **Petits fichiers stockés en interne dans le MFT NTFS.** Les fichiers d'environ 700 octets ou moins sont stockés directement dans l'entrée MFT (attribut \$DATA résident) et ne sont jamais affectés par TRIM, puisqu'ils ne sont jamais écrits dans des secteurs de données séparés.
- **Fragments dans des blocs encore partiellement utilisés.** Tant qu'un bloc NAND contient au moins une page valide, il ne peut pas être effacé en entier — donc les pages invalides du même bloc survivent jusqu'à ce que le garbage collector les déplace.

2.8 Comment vérifier si TRIM est actif

```
Windows : fsutil behavior query DisableDeleteNotify
(DisableDeleteNotify=0 -> TRIM activé,
DisableDeleteNotify=1 -> TRIM désactivé)

Linux : cat /sys/block/sdX/queue/discard_max_bytes
(0 = pas de support TRIM,
valeur non nulle = TRIM disponible)

Pour voir si fstrim s'exécute automatiquement :
systemctl status fstrim.timer

macOS : system_profiler SPSerialATADataType | grep -i 'TRIM Support'
(Yes / No)
```

PARTIE I — FONDATIONS PHYSIQUES

Chapitre 3

Systèmes de fichiers : la carte du trésor

3.1 Pourquoi le système de fichiers décide

Le système de fichiers (FS) est la couche logicielle qui transforme un bloc-périphérique brut (une succession de secteurs) en arborescence de fichiers et dossiers nommés. Il maintient pour cela des **structures internes** qui font le lien entre nom de fichier, métadonnées (taille, dates, permissions) et localisation physique des données sur le support.

Quand on supprime un fichier, le FS modifie typiquement deux ou trois de ces structures internes. Le contenu du fichier lui-même n'est pas touché. C'est cette dissymétrie qui rend possible la récupération logique : **les données brutes survivent à la suppression de leur entrée dans la « carte »**.

Mais chaque FS gère cette « carte » différemment. Certains conservent beaucoup de traces (NTFS, avec son journal \$LogFile, est très bavard), d'autres peu (ext4 libère les inodes et extents assez agressivement). Cela se traduit en chances de récupération sensiblement différentes.

3.2 FAT32 et exFAT : la simplicité

Ce sont les FS les plus simples encore utilisés massivement, principalement sur les supports amovibles (clés USB, cartes SD, appareils photo, dashcams). FAT32 est limité à 4 Go par fichier ; exFAT (2006, Microsoft) lève cette limite et est devenu le standard inter-plateforme pour le stockage de masse.

Structure : un **boot sector** au début, une ou deux **tables FAT** qui décrivent la chaîne de clusters de chaque fichier, et le reste du volume en zone de données. Chaque fichier dans un répertoire est décrit par une entrée de 32 octets contenant nom court, attributs et premier cluster.

À la suppression :

- Le premier caractère de l'entrée de répertoire est remplacé par 0xE5, marquant l'entrée comme supprimée.
- Les clusters dans la table FAT sont marqués libres (remis à zéro).
- Le contenu des clusters lui-même reste intact jusqu'à réécriture.

La récupération est en général très efficace sur FAT/exFAT, surtout pour les fichiers contigus (peu de fragmentation, ce qui est fréquent sur des cartes SD utilisées en mode séquentiel). Limite principale : la première lettre du nom est perdue. Les outils mettent souvent un caractère générique (X) à la place.

3.3 NTFS : la richesse forensique

NTFS (*New Technology File System*, Microsoft, 1993) est le FS standard de Windows depuis Windows NT. C'est techniquement le FS le plus généreux en métadonnées résiduelles, ce qui en fait le plus récupérable d'un point de vue logique.

Sa structure centrale est la **Master File Table (\$MFT)** : un fichier spécial qui contient une entrée de 1 024 octets par fichier et par répertoire du volume. Cette entrée comprend :

- Un en-tête (42 premiers octets) avec un drapeau indiquant si l'entrée est utilisée ou supprimée.
- Un attribut `$STANDARD_INFORMATION` avec les quatre timestamps MACB (Modified, Accessed, Created, Birth/MFT change).
- Un attribut `$FILE_NAME` avec le nom et une référence au répertoire parent.
- Un attribut `$DATA` qui contient soit le contenu du fichier directement (s'il fait moins de ~700 octets, on parle d'attribut *résident*), soit une liste de *runs* pointant vers des clusters de données.

À la suppression, NTFS se contente de basculer le drapeau de l'en-tête à « supprimé » et de marquer les clusters comme libres dans `$Bitmap`. L'entrée MFT elle-même reste, avec tous ses attributs. Le journal `$LogFile` conserve souvent la trace des dernières opérations, et le journal `$UsnJrnl` (activé par défaut sur les Windows modernes) répertorie chaque création, modification ou suppression avec horodatage.

Outils NTFS — Pour le forensique NTFS, deux outils dominant : MFTECmd d'Eric Zimmerman (parse le `$MFT` en CSV exploitable dans Timeline Explorer) et `dfir_ntfs` de Maxim Suhanov. Tous deux savent récupérer des métadonnées dans le **MFT slack space** — les zones non utilisées à l'intérieur des entrées MFT, qui contiennent souvent des fragments d'anciennes entrées (article de référence : Sygnia, *The Forensic Value of MFT Slack Space*, 2025).

3.4 ext4 : la spécificité Linux

ext4 (2008) est le FS par défaut de la plupart des distributions Linux. Trois éléments structurels clés :

- Le **superblock**, qui contient les paramètres globaux du FS (nombre total d'inodes, taille des blocs, etc.).
- La **table d'inodes**, qui contient une structure par fichier/dossier avec ses métadonnées.
- Les **extents** : ext4 ne décrit pas les blocs de données fichier par fichier (comme ext3 le faisait avec ses pointeurs indirects), mais par plages contigües (extents). Plus efficace pour les gros fichiers, mais plus destructeur à la suppression.

À la suppression d'un fichier sur ext4 :

1. L'inode est marqué libre dans le bitmap d'inodes.
2. Les extents sont libérés dans le bitmap de blocs.
3. Dans l'inode lui-même, ext4 efface partiellement les pointeurs vers les blocs (contrairement à ext3 qui les conservait). C'est ce qui rend la récupération plus difficile sur ext4 que sur ext3.

L'outil de référence est **extundelete** (open source, extundelete.sourceforge.net), qui exploite le journal ext4 pour retrouver les anciennes versions des inodes supprimés avant que le journal lui-même ne les réécrive. Quand cela échoue, **debugfs** (inclus dans `e2fsprogs`) permet d'examiner manuellement la structure :

```
$ sudo debugfs /dev/sda1
debugfs: lsdel # liste les inodes récemment supprimés
debugfs: stat <12345> # détails de l'inode 12345
debugfs: dump <12345> /chemin/fichier.bin
```

Attention — Sur ext4, si vous venez de supprimer quelque chose d'important, remontez immédiatement la partition en lecture seule avant toute autre action : `sudo mount -o remount,ro /dev/sdXY`. Toute écriture, même un simple log système, peut réutiliser les inodes ou les blocs libérés.

3.5 APFS : copy-on-write et snapshots

APFS (*Apple File System*, 2017) a remplacé HFS+ sur macOS, iOS et iPadOS. C'est un FS moderne qui repose sur deux principes :

- **Copy-on-write.** Toute modification écrit ailleurs et met à jour les pointeurs ; les anciennes versions ne sont jamais directement écrasées. Cela garantit la cohérence en cas de coupure.
- **Snapshots.** APFS sait conserver des images instantanées d'état antérieur du volume, à coût marginal presque nul (puisque le copy-on-write conserve déjà les anciens blocs). Time Machine sur macOS s'appuie intensivement sur ce mécanisme.

Conséquence pour la récupération : sur un APFS non chiffré, des fichiers supprimés il y a des semaines peuvent être présents dans un snapshot local. Outils comme R-Studio, UFS Explorer et Disk Drill exploitent ces snapshots.

Le mur, c'est FileVault. Activé par défaut sur les Mac modernes avec puce Apple Silicon, FileVault chiffre tout le volume avec AES-256, et la clé est protégée par le mot de passe utilisateur et la *Secure Enclave*. Sans le mot de passe, la donnée physique sur le support n'est qu'un flux pseudo-aléatoire.

3.6 Btrfs et ZFS : robustesse maximale

Btrfs (Oracle, intégré au noyau Linux depuis 2009) et ZFS (Sun Microsystems, 2006, désormais OpenZFS) sont deux FS de la famille « copy-on-write + checksums + snapshots », pensés pour la résilience à grande échelle. On les retrouve principalement sur les NAS (Synology, QNAP), les serveurs Linux et les appliances de stockage (TrueNAS).

- **Checksums.** Chaque bloc est protégé par une somme de contrôle. La détection silencieuse de corruption (*bit rot*) est intégrée.
- **Self-healing.** Sur un volume miroir ou RAID-Z, ZFS peut réécrire automatiquement un bloc corrompu à partir d'une copie saine.
- **Snapshots.** Comme APFS, à coût quasi nul, et exploités par les NAS modernes pour offrir aux utilisateurs des sauvegardes ponctuelles.

Pour la récupération : ces FS sont en pratique très peu vulnérables à la perte par corruption logique simple ; ils le sont par contre à la **fragmentation extrême** (le copy-on-write fragmente naturellement le contenu au fil des modifications) et à la complexité de leurs structures internes, ce qui rend le data carving classique très inefficace. L'approche la plus productive sur ces FS est presque toujours **via les snapshots**, pas via le carving.

3.7 Tableau de synthèse

FS	Plateformes	Comportement à la suppression	Qualité de récupération logique
FAT32	USB, SD, anciens systèmes	Entrée marquée 0xE5, FAT remise à zéro	Très bonne (perte de la première lettre du nom)
exFAT	USB, SD, cartes haute capacité	Identique à FAT32, capacité étendue	Très bonne
NTFS	Windows	MFT entry marquée supprimée, \$LogFile et \$LogFile2 conservés	Excellente pour les fichiers ; nulle pour les métadonnées résiduelles
ext4	Linux	Inode libéré, extents effacés agressivement	Moyenne — extundelete + journal aident dans la fenêtre courte
APFS	macOS, iOS	Copy-on-write, snapshots conservés selon politique	Excellente via snapshots ; nulle si FileVault sans clé
Btrfs / ZFS	NAS, serveurs Linux	Copy-on-write, snapshots, checksums	Excellente via snapshots ; complexe sinon

Pourquoi pas de pourcentages — Aucun pourcentage précis n'a sa place dans ce tableau, contrairement à ce qu'on lit souvent sur internet. La récupération réelle dépend de paramètres multiples : temps écoulé depuis la suppression, activité du système entre temps, fragmentation des fichiers, type de support (HDD ou SSD avec TRIM), état du journal. Méfiez-vous des tableaux qui annoncent « 87,3 % de taux de récupération sur NTFS » : personne ne sait mesurer cela rigoureusement.

Partie II

Diagnostic

Avant d'agir, comprendre. Deux chapitres : un panorama chiffré des causes de perte de données en 2025-2026, et une méthode de triage pour décider si vous êtes face à un cas logique ou physique, et si vous pouvez intervenir vous-même ou si l'envoi en laboratoire s'impose.

PARTIE II — DIAGNOSTIC

Chapitre 4

Causes de perte : les chiffres 2025-2026

4.1 Quatre familles de causes

On classe les pertes en quatre catégories qui appellent des réponses très différentes :

- **Humaines et logiques** : suppression accidentelle, formatage, mauvaise manipulation, erreur de configuration. Le support est sain ; les données sont logiquement inaccessibles mais physiquement présentes.
- **Cyber** : ransomware, malware destructeur, wiper, suppression malveillante. Le ransomware ajoute une couche de chiffrement ; les wipers (NotPetya par exemple) détruisent réellement.
- **Matérielles** : panne mécanique (HDD), défaillance électronique (PCB grillé), usure NAND (SSD).
- **Environnementales** : incendie, inondation, surtension, vol, destruction physique.

4.2 Le ransomware, la nouvelle norme

Le rapport Verizon DBIR 2025 est la référence statistique sur les compromissions de données. Sur la période couverte (novembre 2023 à octobre 2024), Verizon a analysé plus de 22 000 incidents et 12 195 compromissions confirmées. Les chiffres principaux :

Indicateur	Valeur 2024 (DBIR 2025)	Tendance
Ransomware dans les breaches	44 %	+37 % vs DBIR 2024
Ransomware dans les breaches PME	88 %	Aggravation des inégalités
Ransomware dans les breaches grandes entreprises	30 %	Stable
Identifiants volés (vecteur initial)	22 %	Toujours n° 1
Vulnérabilités exploitées (vecteur initial)	20 %	+34 %
Implication d'un tiers (supply chain)	30 %	Double vs DBIR 2024 (15 %)
Médiane des rançons payées	115 000 \$	En baisse (150 k\$ en 2023)
Refus de payer la rançon	64 % des victimes	vs 50 % deux ans plus tôt

Source : Verizon Business, 2025 Data Breach Investigations Report, publié le 23 avril 2025.

Deux lectures opposées du même rapport. La pessimiste : le ransomware est devenu un mode opératoire dominant, particulièrement dévastateur pour les PME qui en sont victimes dans neuf cas sur dix. L'optimiste : la médiane des rançons baisse, deux victimes sur trois refusent désormais de payer. Les politiques publiques (formation, exigences cyber) commencent à porter — modestement.

Précision méthodologique — Le DBIR groupe désormais les rançongiciels avec extorsion pure (« pure extortion », sans chiffrement, juste exfiltration et menace de divulgation). Cela explique en partie la hausse — les deux catégories étaient comptées séparément auparavant. La tendance reste forte même en tenant compte de ce regroupement.

4.3 L'erreur humaine, toujours majoritaire

L'élément humain (au sens large : erreur, abus de privilège, ingénierie sociale) reste impliqué dans une part dominante des compromissions. Le DBIR 2025 chiffre cela à 60 % sur l'ensemble des breaches étudiés. Les définitions varient d'un rapport à l'autre — certains atteignent 95 % en incluant toute action humaine en amont de l'incident — mais l'ordre de grandeur est constant.

Pour la récupération de données spécifiquement (et pas pour les compromissions au sens cyber), les causes humaines les plus fréquemment rencontrées par les laboratoires sont :

- Suppression accidentelle de fichiers ou de répertoires (y compris vidage de corbeille).
- Formatage par erreur (souvent au moment de l'installation d'un OS).
- Écrasement de fichiers par mauvaise manipulation (copie d'un dossier dans le mauvais sens, par exemple).
- Suppression massive par script ou commande (rm -rf mal placé, drop database, etc.).
- Perte de mot de passe ou de clé de chiffrement.

4.4 La fiabilité matérielle moyenne

Pour les pannes matérielles sur HDD, Backblaze reste la référence publique avec son rapport *Drive Stats 2025* (décembre 2025 / février 2026). Quelques chiffres dignes d'attention :

- Sur 337 192 disques de production fin 2025, AFR annuel global de **1,36 %**.
- AFR cumulé sur la vie des disques (lifetime) : **1,30 %**.
- Modèles « zéro panne » du trimestre : certains disques Seagate 8 To (ST8000NM000A) n'ont eu qu'une panne en cinq trimestres consécutifs.
- Modèles à très haut AFR (outliers) : un disque Toshiba MG09 sur Q3 2025 a affiché 16,9 % d'AFR, expliqué après enquête par une opération de déploiement firmware qui faussait le décompte.

Pour les SSD, aucun rapport public comparable n'existe à grande échelle. Backblaze publie quelques statistiques sur ses SSD de boot mais le parc est trop petit pour être généralisable. L'endurance théorique des SSD modernes (TBW garanti par les constructeurs) couvre normalement 5 à 10 ans d'usage consumer ; la panne effective est souvent due à un contrôleur défaillant ou un firmware corrompu, pas à l'usure des cellules.

4.5 Synthèse

En 2025-2026, le paysage des pertes de données se résume ainsi :

1. L'**erreur humaine** reste la cause numéro un en volume, particulièrement chez les particuliers et les PME.
2. Le **ransomware** a explosé pour devenir la première cause en termes d'impact financier — il est désormais présent dans près d'une compromission sur deux en entreprise.
3. Les **pannes matérielles** sont en lente diminution sur HDD (meilleure qualité, AFR sous les 1,5 %) ; sur SSD, elles sont moins fréquentes mais souvent plus catastrophiques quand elles surviennent.
4. Les **incidents environnementaux** représentent quelques pourcents des cas, mais leur coût peut être colossal (datacenter détruit, perte totale).

PARTIE II — DIAGNOSTIC

Chapitre 5

Diagnostic et triage

5.1 Pourquoi le diagnostic est l'étape critique

Le diagnostic décide de tout : du choix entre intervention logicielle ou physique, du choix entre tentative personnelle ou envoi en laboratoire, du coût et du délai prévisibles, et souvent du résultat lui-même. Un mauvais diagnostic conduit à des actions inadaptées qui aggravent la situation.

Attention — Tant que le diagnostic n'est pas posé, ne touchez à rien. Surtout, ne lancez aucun outil de récupération « pour voir » : beaucoup d'entre eux écrivent sur le support source dès qu'on les installe ou dès le premier scan.

5.2 Procédure de triage en cinq étapes

1. **Observer.** Le support est-il alimenté ? Émet-il du bruit ? Lequel ? Est-il chaud ? Le BIOS/UEFI le détecte-t-il ? L'OS l'affiche-t-il dans les outils standards (Gestionnaire de disques Windows, Utilitaire de disque macOS, lsblk Linux) ? Notez tout sans rien modifier.
2. **Classer.** À partir des observations, classer en : *physique* (non détecté, bruits anormaux, surchauffe), *logique* (détecté mais inaccessible, fichiers manquants, FS corrompu), ou *hybride* (détection intermittente, lecture partielle).
3. **Évaluer l'enjeu.** Les données sont-elles irremplaçables ? Ont-elles une valeur judiciaire, médicale, professionnelle ? Existent-elles ailleurs (sauvegarde, cloud) ? L'urgence est-elle réelle ?
4. **Décider du chemin.** Selon la combinaison (type / enjeu / compétences disponibles), choisir : intervention personnelle, intervention par un technicien IT généraliste, ou envoi en laboratoire spécialisé.
5. **Documenter.** Si l'enjeu peut devenir judiciaire, photographier l'état du support, noter les numéros de série, tracer chaque manipulation. C'est la chaîne de custody (voir chapitre 14).

5.3 Symptômes typiques et leur signification

Symptôme observé	Diagnostic probable	Action immédiate
Clics répétés sur HDD ('tick, tick, tick')	Têtes de lecture HS — physique	Éteindre immédiatement, ne pas rebrancher
HDD silencieux, non détecté	PCB ou moteur HS — physique	Éteindre, envoi labo
HDD détecté mais grandes lenteurs	Secteurs défectueux — physique	Évaluation d'urgence avec ddrescue
SSD non détecté du tout	Contrôleur HS — physique	Éteindre, envoi labo (JTAG/chip-off)
SSD détecté, capacité bizarre (8 Mo, 0 Mo)	Firmware corrompu — physique	Envoi labo
Partition manquante, table de partition corrompue	Logique	Image ddrescue puis TestDisk
Fichiers supprimés par erreur	Logique — fenêtre courte sur SSD	Débrancher, ne rien écrire dessus
Volume RAW (NTFS/exFAT corrompu)	Logique	Image puis R-Studio/UFS Explorer
Fichiers chiffrés avec extensions inconnues	Ransomware	Voir chapitre 12
Demande de mot de passe sur tout le disque	BitLocker/FileVault/LUKS	Trouver la clé de récupération

5.4 Quand passer en laboratoire

Trois critères, dont au moins un suffit :

- Le support n'est pas détecté, ou émet des bruits anormaux → physique → labo.
- Les données sont irremplaçables et critiques (médical, judiciaire, professionnel à fort enjeu) → labo, même si le cas semble simple.
- Vous avez déjà tenté quelque chose qui a aggravé la situation → arrêtez, et laissez un pro évaluer ce qui reste récupérable.

Quand passer par un labo, comment choisir ? Quelques critères :

- Salle blanche ISO 5 certifiée (demandez le certificat ISO 14644-1).
- Diagnostic gratuit et paiement au résultat — standard de marché en 2026.
- Confidentialité écrite (NDA), surtout pour les cas professionnels.
- Réputation vérifiable (avis indépendants, publications techniques, mentions dans la presse spécialisée). Ontrack, Kroll, DriveSavers, SalvageData, Gillware, Secure Data Recovery font partie des références internationales souvent citées.

Signes de laboratoire douteux — Si on vous demande un règlement intégral avant diagnostic, ou qu'on vous promet un taux de succès garanti, partez. Aucun laboratoire sérieux ne promet une récupération avant d'avoir ouvert le support.

Partie III**Méthodes**

Six chapitres qui constituent le cœur opérationnel du livre : comment imager un support (la fondation de toute autre opération), comment analyser la couche logique, comment sculpter des fichiers à partir de données brutes (carving), comment intervenir physiquement sur un disque dur et sur un SSD, et comment gérer les configurations RAID.

PARTIE III — MÉTHODES

Chapitre 6

Imagerie sécurisée

6.1 Le principe : ne jamais travailler sur l'original

L'imagerie est l'étape qui sépare la récupération amateur de la récupération professionnelle. Elle consiste à créer une copie bit à bit du support source vers un fichier image, puis à travailler exclusivement sur cette image. L'original est mis de côté, à l'abri.

Trois raisons :

1. **Sauvegarder ce qui peut l'être.** Un support qui tombe en panne en fait souvent plus pendant les heures suivantes. Chaque minute compte ; la première lecture peut être la dernière qui réussit.
2. **Travailler sereinement.** Sur l'image, on peut faire autant d'essais qu'on veut. Si un outil corrompt quelque chose, on en crée une autre copie.
3. **Préserver la preuve.** En forensique, l'original est scellé avec son hash de référence ; tout travail se fait sur une copie hashée elle aussi. C'est ce qui rend la procédure recevable en justice (norme ISO 27037).

6.2 ddrescue : l'outil de référence

GNU ddrescue (paquet `gddrescue` sous Debian, Ubuntu, et la plupart des distributions Linux) est l'outil open source de référence. Sa supériorité sur `dd` classique tient à trois éléments :

- Un fichier de carte (*mapfile*) qui enregistre précisément les zones déjà copiées, à recopier, lentes ou échouées. Il permet de reprendre exactement où on s'était arrêté, et de programmer des passes ciblées.
- Une stratégie de lecture en plusieurs passes : rapide d'abord (on saute les zones lentes pour récupérer le facile), agressive ensuite (on revient sur les zones difficiles).
- Une gestion fine des secteurs en erreur, avec nombre maximal de tentatives configurable.

Workflow type

Identifier le périphérique avec précision avant tout autre chose. Une erreur de lettre, et c'est un disque sain qui est écrasé.

```
$ lsblk -o NAME,SIZE,MODEL,SERIAL,TRAN
NAME SIZE MODEL SERIAL TRAN
sda 500G ST500LM030-2E717D WCC6Y0L1234 sata
sdb 2T WDC_WD20EZRZ WD-WCC4M1234 usb
...

# Le serial number permet de confirmer qu'on vise le bon disque.
```

Première passe, rapide : on copie ce qui se lit facilement, on saute le reste.

```
$ sudo ddrescue -f -n -d /dev/sdX /chemin/image.img /chemin/image.map

-f : autorise la sortie vers un device
-n : 'no-scrape' : ne s'attarde pas sur les zones difficiles
-d : accès direct au device (bypass cache OS)
```

Deuxième passe, ciblée sur les zones difficiles avec quelques tentatives :

```
$ sudo ddrescue -f -d -r3 /dev/sdX /chemin/image.img /chemin/image.map  
  
-r3 : jusqu'à 3 retries par secteur en erreur
```

Sur un disque qui claque (têtes HS), on évite d'aller plus haut que -r3 : chaque tentative supplémentaire est une occasion de plus d'endommager la surface. Sur un disque dont le problème est juste électronique (PCB partiellement défaillant, USB instable), on peut aller jusqu'à -r10 sans risque physique supplémentaire.

Option utile sur certains cas particulièrement abîmés : passe en lecture inverse, qui aide quand l'ordre de présentation des secteurs joue (par exemple si la tête se positionne mal sur certains cylindres) :

```
$ sudo ddrescue -f -d -R -r3 /dev/sdX /chemin/image.img /chemin/image.map  
  
-R : 'reverse' : lit de la fin vers le début
```

Une fois l'image obtenue

Calculer un hash de l'image. C'est la référence d'intégrité à laquelle on comparera plus tard pour prouver qu'on n'a rien modifié :

```
$ sha256sum /chemin/image.img > /chemin/image.img.sha256
```

Pour explorer l'image en lecture seule sans risque, on la monte via un loop device :

```
$ sudo losetup --read-only --find --show /chemin/image.img  
/dev/loop0  
  
# Sur ext4 : noload empêche le rejeu de journal (qui écrirait)  
$ sudo mount -o ro,noload /dev/loop0 /mnt/recup  
  
# Sur NTFS via ntfs-3g :  
$ sudo mount -t ntfs-3g -o ro,norecover /dev/loop0 /mnt/recup  
  
# Démontez et libérez le loop quand on a fini :  
$ sudo umount /mnt/recup  
$ sudo losetup -d /dev/loop0
```

Attention — Stockez le mapfile et l'image sur un support distinct du source. Si l'image est sur la même clé USB que le support en panne (cas vu en labo : l'utilisateur n'avait que ça sous la main), et que cette clé tombe, on a tout perdu. Toujours sur un disque sain, rapide et avec assez de place.

6.3 Alternatives à ddrescue

FTK Imager (AccessData/Exterro, gratuit). Outil Windows très populaire en forensique. Crée des images au format E01 (EnCase) ou DD brut, calcule MD5/SHA1/SHA256 automatiquement, supporte les write blockers hardware.

Guymager (Linux, GUI). Interface graphique pour la création d'images forensiques. Plus pratique que ddrescue pour les utilisateurs occasionnels mais moins flexible sur les cas difficiles.

dc3dd (DoD Cyber Crime Center). Variante de dd améliorée pour la forensique : hash à la volée, journalisation, validation. Utile quand ddrescue n'est pas adapté (image linéaire d'un support sain pour preuve).

Outils hardware : PC-3000 Disk Imager (ACE Lab), DeepSpar Disk Imager, Atola Insight Forensic. Ces solutions hardware gèrent les supports gravement endommagés mieux que les outils purement logiciels (contrôle direct du contrôleur, gestion fine des resets, masquage des têtes HS sur HDD). Réservés aux laboratoires en raison du coût (plusieurs milliers à dizaines de milliers d'euros).

6.4 Write blockers : la sûreté maximale

Pour les cas forensiques (preuve devant tribunal), on intercale entre le support source et la machine d'analyse un **write blocker** matériel : un dispositif qui laisse passer les lectures mais bloque physiquement toute écriture. Cela garantit qu'aucun bug logiciel ne peut modifier la source. Les marques de référence : Tableau (Guidance Software/OpenText), WiebeTech (CRU).

Pour un usage non forensique mais prudent, le write blocker logiciel disponible via les paramètres BIOS/UEFI (« write protect ») ou via des outils comme blockdev --setro sous Linux suffit en pratique.

PARTIE III — MÉTHODES

Chapitre 7

Analyse logique et réparation FS

7.1 Le principe : reconstruire la carte avant le contenu

Une fois l'image obtenue, l'analyse logique cherche à **réparer ou interpréter les structures du système de fichiers** présentes sur l'image, plutôt que de chercher directement des fichiers dans les données brutes. C'est presque toujours plus efficace que le carving : on récupère non seulement le contenu, mais aussi les noms, les dates, l'arborescence.

L'ordre logique standard est :

1. Réparation de la table de partition (si manquante ou corrompue).
2. Réparation des structures du FS (MFT pour NTFS, inodes pour ext4, snapshot APFS, etc.).
3. Extraction des fichiers présents (utilisateurs et supprimés).
4. Si la couche logique reste inexploitable, basculer en carving (chapitre 8).

7.2 TestDisk : la table de partition

TestDisk (Christophe Grenier, CGSecurity, gratuit, open source) est l'outil de référence pour la réparation de tables de partition MBR et GPT, et pour la récupération de partitions effacées. Il fonctionne sous Windows, macOS et Linux, en mode texte (ncurses).

Workflow type sur une image qui a perdu sa table :

1. Lancer `testdisk /chemin/image.img`.
2. Choisir « None » pour la création de log (sur une image lecture seule).
3. Sélectionner le disque/image, puis le type de table (Intel/PC pour MBR, EFI GPT pour GPT moderne).
4. Lancer « Analyse » puis « Quick Search ». TestDisk scanne le support et propose les partitions trouvées.
5. Si Quick Search ne suffit pas, lancer « Deeper Search » (lent mais exhaustif).
6. Vérifier que les partitions trouvées sont bien les bonnes (taille, type), puis écrire la nouvelle table (« Write »).

Écrire sur l'image — Sur une *image* de récupération, on peut écrire sans scrupule — l'original physique n'est pas touché. C'est une raison de plus pour toujours imager d'abord.

7.3 NTFS : MFT, \$LogFile, \$UsnJrnl

Pour analyser un volume NTFS, plusieurs outils complémentaires :

- **R-Studio** et **UFS Explorer** (commerciaux) sont les références pour la reconstruction NTFS — ils savent réparer une MFT partiellement endommagée, remonter l'arborescence à partir des entrées supprimées, exploiter le journal.

- **MFTECmd** (Eric Zimmerman, gratuit) parse le fichier \$MFT en CSV exploitable. Permet de lister tous les fichiers et dossiers du volume — y compris ceux supprimés dont l'entrée MFT existe encore.
- **LogFileParser** et **UsnJrnl2Csv** (Eric Zimmerman) exploitent les journaux pour reconstituer la chronologie des opérations récentes.
- **The Sleuth Kit** (TSK, gratuit, open source) et son interface **Autopsy** offrent un cadre complet d'analyse forensique multi-FS, dont NTFS.

Exemple de chaîne d'analyse NTFS en ligne de commande sous Linux :

```
# Monter l'image en lecture seule
$ sudo losetup --read-only --find --show /chemin/image.img
$ sudo mount -t ntfs-3g -o ro,norecover /dev/loop0 /mnt/recup

# Avec The Sleuth Kit : lister tous les fichiers, y compris supprimés
$ fls -r -p /chemin/image.img > fichiers.txt
# Les fichiers supprimés sont marqués '*' au début de la ligne

# Récupérer un fichier supprimé par son inode
$ icat /chemin/image.img 12345 > fichier_recup.bin
```

7.4 ext4 : extundelete, debugfs, ext4magic

Trois outils principaux :

- **extundelete** exploite le journal ext4 pour retrouver les anciennes versions des inodes supprimés. Restaure dans un répertoire dédié RECOVERED_FILES/.
- **debugfs** (inclus dans e2fsprogs) fournit un accès interactif aux structures internes. Permet le diagnostic fin (lister les inodes récemment supprimés, dumper un inode précis).
- **ext4magic** combine les deux approches et offre une interface de plus haut niveau.

Exemple :

```
# Volume démonté ou monté en RO !
$ sudo umount /dev/sdb1 # si monté

# Restaurer un fichier précis dont on connaît le nom
$ sudo extundelete --restore-file 'home/user/important.pdf' /dev/sdb1

# Restaurer tout ce qui est récupérable
$ sudo extundelete --restore-all /dev/sdb1

# Avec debugfs, lister les inodes supprimés (Linux)
$ sudo debugfs /dev/sdb1
debugfs: lsdel
Inode Owner Mode Size Blocks Time deleted
1234 1000 100644 524288 128 Sun May 4 14:32:11 2026
debugfs: dump <1234> /tmp/recovered
debugfs: quit
```

7.5 APFS : exploiter les snapshots

Sur APFS non chiffré, la priorité absolue est de chercher les snapshots locaux. Time Machine en crée régulièrement et les conserve un certain temps même quand on est éloigné du disque externe Time Machine.

```
# Lister les snapshots APFS sur un Mac vivant
$ tmutl listlocalsnapshots /
```

```
# Lister les snapshots sur un volume monté en lecture seule
$ diskutil apfs listSnapshots /Volumes/data
```

R-Studio et **UFS Explorer Professional** savent exploiter les snapshots APFS — y compris à partir d'une image brute. Ils sont préférables à l'écriture manuelle de scripts pour les cas non triviaux.

Sur APFS chiffré (FileVault), pas de mystère : sans la clé (mot de passe utilisateur ou clé de récupération à 24 caractères stockée par Apple à l'activation, ou clé institutionnelle pour les Mac d'entreprise), aucune récupération possible. Vérifier auprès de l'utilisateur — beaucoup ont leur clé de récupération dans iCloud sans le savoir.

7.6 Btrfs et ZFS : snapshots et listes

Sur Btrfs, les snapshots sont gérés au niveau des **sous-volumes**. Pour les NAS Synology (qui utilisent Btrfs avec snapshots automatiques), la console DSM permet de revenir à un état antérieur en quelques clics. En ligne de commande sur le NAS ou une machine Linux :

```
# Lister les snapshots Btrfs
$ sudo btrfs subvolume list /volume1

# Restaurer un fichier depuis un snapshot
$ cp /volume1/.snapshots/123/fichier.pdf /volume1/fichier.pdf
```

Sur ZFS, c'est encore plus direct :

```
# Lister les snapshots ZFS
$ sudo zfs list -t snapshot

# Roll back à un snapshot précis (perte des modifs ultérieures !)
$ sudo zfs rollback pool/dataset@snapshot_name

# Plus prudent : juste accéder en lecture seule au snapshot
$ ls /pool/dataset/.zfs/snapshot/snapshot_name/
```

PARTIE III — MÉTHODES

Chapitre 8

Data carving en profondeur

8.1 Quand carver

Le data carving (« sculpture de fichiers ») est l'opération qui reconstitue des fichiers en cherchant leurs signatures binaires dans les données brutes du support, **sans utiliser les structures du système de fichiers**. C'est l'opération de dernier recours quand la couche logique est trop endommagée pour être interprétée.

Cas typiques :

- Formatage complet (toute la structure FS a été réécrite, le contenu des fichiers est toujours là).
- Image brute issue d'un chip-off (pas de FS du tout, juste de la NAND ré-assemblée).
- Réinstallation d'OS qui a écrit son nouveau système sur l'ancien (l'ancienne structure est partiellement écrasée).
- Support tellement corrompu que les outils logiques ne trouvent rien.

8.2 Quatre niveaux de sophistication

Niveau 1 — Signature simple (header / footer)

L'approche historique. On scanne le support à la recherche d'un *magic number* caractéristique du début du format de fichier, puis on lit jusqu'au *magic number* de fin (s'il existe) ou jusqu'à une taille maximale fixée. Quelques exemples de signatures classiques :

Format	Header (hex)	Footer / fin
JPEG (JFIF)	FF D8 FF E0	FF D9
JPEG (Exif)	FF D8 FF E1	FF D9
PNG	89 50 4E 47 0D 0A 1A 0A	49 45 4E 44 AE 42 60 82
PDF	25 50 44 46 ("%PDF")	25 25 45 4F 46 ("%EOF")
ZIP / DOCX / XLSX	50 4B 03 04	Variable (central directory)
MP4 / MOV	(offset 4) 66 74 79 70	Pas de footer fixe
RAR (v5+)	52 61 72 21 1A 07 01 00	Variable
SQLite	53 51 4C 69 74 65 20 66 6F 72 6D 61 74 20 33 09	

Très efficace sur des fichiers **non fragmentés** et de taille modérée. Échoue dès qu'un fichier est fragmenté (le carver suit aveuglément les blocs contigus et finit par incorporer des données d'un autre fichier ou des zéros). Outils typiques : **PhotoRec**, **Foremost**, **Scalpel**.

Niveau 2 — Carving sémantique (structure-aware)

On exploite la structure interne du format pour valider et réassembler. Pour un PDF, c'est la table de références croisées (*xref*) qui pointe vers chaque objet ; pour un ZIP, c'est le *central directory* en fin de fichier. Cela permet de reconstituer des fichiers même fragmentés, à condition de retrouver tous les morceaux.

Les outils forensiques commerciaux (**Belkasoft X**, **X-Ways Forensics**, **Magnet AXIOM**) intègrent ce niveau pour les formats courants. Les outils open source (**Bulk Extractor**) sont bons sur certains formats spécifiques.

Niveau 3 — Analyse d'entropie

On calcule l'entropie de Shannon par bloc pour distinguer trois régimes : zones très ordonnées (texte, code, structures), zones modérément entropiques (images compressées, archives), zones à entropie maximale (données chiffrées ou aléatoires). Cela permet de cibler le carving sur les zones intéressantes et d'éviter d'extraire des fragments de données chiffrées qui ne donneraient rien.

Niveau 4 — Apprentissage automatique

Les approches récentes (2023-2026) utilisent des réseaux de neurones pour classifier des fragments et tenter de les réassembler. Projets de recherche notables : **Carve-DL** (université du Texas, 2023), travaux de la Forensic Computing Research Group à Mannheim. Belkasoft X et Magnet AXIOM annoncent des modules ML depuis 2024-2025. Les résultats sont prometteurs sur certains formats (images compressées, documents Office) mais restent en pratique limités. La littérature publique ne donne pas de taux de succès comparables et reproductibles.

8.3 PhotoRec : le standard open source

PhotoRec (CGSecurity, fourni avec TestDisk) est l'outil de carving open source le plus utilisé. Multiplate-forme, supporte plus de 480 signatures de fichiers, et est conçu pour fonctionner sur des images comme sur des supports directs.

```
# Lancer PhotoRec sur une image
$ sudo photorec /chemin/image.img

# Lancer sur un device direct (à éviter quand on a une image)
$ sudo photorec /dev/sdb
```

L'interface est en mode texte (ncurses). Étapes :

1. Choisir le support / l'image.
2. Sélectionner le type de table de partition (Intel/PC en général).
3. Choisir la partition (ou « Whole disk » si la table est détruite).
4. Choisir le système de fichiers d'origine (ext, NTFS, FAT) pour optimiser l'algorithme — ou « Other » si inconnu.
5. **File Opt** : sélectionner les types de fichiers à chercher. Désactivez ce dont vous n'avez pas besoin, sinon vous allez obtenir des millions de fichiers parasites.
6. Choisir le répertoire de sortie (**jamais sur le support source**).
7. Laisser tourner. PhotoRec est lent : compter plusieurs heures pour 1 To.

Attention — PhotoRec ne récupère **jamais** les noms de fichiers d'origine — il les renomme en f0000001.jpg, f0000002.pdf, etc. Il faut prévoir un tri manuel ou scripté après l'extraction. Pour un usage forensique sérieux, préférer une approche FS-aware (chapitre 7) quand c'est possible.

8.4 Scalpel et Foremost

Foremost (créé par l'Air Force Office of Special Investigations) et **Scalpel** (fork de Foremost avec améliorations de performance) sont deux outils de carving en ligne de commande qui s'appuient sur un fichier de configuration listant les signatures à chercher.

```
# Foremost - carving rapide avec config par défaut
$ sudo foremost -i image.img -o output/

# Avec types spécifiques
$ sudo foremost -t jpg,pdf,doc -i image.img -o output/

# Scalpel - utilise scalpel.conf qu'on édite avant
$ sudo scalpel -c /etc/scalpel/scalpel.conf -o output/ image.img
```

8.5 Limites dures du carving

- **Fragmentation.** Sur des FS très fragmentés (ext4 utilisé depuis longtemps, ZFS/Btrfs en copy-on-write), les fichiers sont éclatés en morceaux. Le carving signature simple récupère le premier fragment puis du bruit. Le carving sémantique aide mais ne miracle pas.
- **Chiffrement.** Du contenu chiffré à une entropie maximale et ne ressemble à aucun format de fichier connu. Sans la clé, le carving ne donne rien.
- **Compression.** Un fichier ZIP, MP3 ou JPEG qui a perdu ses premiers blocs est en général irrécupérable même si le reste est intact — la compression est par construction « tout ou rien ».
- **Faux positifs.** Beaucoup de séquences aléatoires ressemblent à des signatures. Sur 1 To de NAND brute issue de chip-off, le carving peut produire des millions de fichiers, dont 99 % sont du bruit.

PARTIE III — MÉTHODES

Chapitre 9

Intervention physique sur HDD

9.1 La salle blanche

Ouvrir un disque dur hors d'une salle blanche est l'erreur la plus coûteuse possible. La couche magnétique des plateaux est épaisse de quelques nanomètres seulement, et les têtes volent à quelques nanomètres au-dessus. Une particule de poussière ambiante (5 à 50 microns typiquement) coincée sous une tête agit comme un caillou sous un train : la couche magnétique est arrachée sur la trajectoire de la tête.

La norme internationale qui définit les classes de salle blanche est **ISO 14644-1**. Elle exprime les limites de concentration de particules par mètre cube en fonction de la taille des particules. Les classes pertinentes pour la récupération de données :

Classe ISO	Particules >= 0,5 micron / m ³	Usage typique
ISO 3	≤ 35	Fabrication semi-conducteurs
ISO 4	≤ 352	Récupération haut de gamme (Secure Data Recovery)
ISO 5 (Class 100)	≤ 3 520	Standard récupération HDD (DriveSavers, SalvageData, Gillware, ACE Data R)
ISO 6 (Class 1000)	≤ 35 200	Trop laxiste pour HDD
ISO 7 (Class 10 000)	≤ 352 000	Zone de préparation, sas
Air ambiant typique	~10 000 000+	À éviter absolument

Sources : ISO 14644-1:2015 ; pages de certification DriveSavers, SalvageData, Gillware, Secure Data Recovery (consultées 2026).

Une salle blanche ISO 5 utilise un flux d'air **laminaire** (unidirectionnel) filtré HEPA ou ULPA. L'air est renouvelé plusieurs centaines de fois par heure. Les techniciens portent des combinaisons intégrales (gants, charlotte, surchaussures), et le matériel est nettoyé à l'isopropanol et passé sous flux laminaire avant entrée.

Coût et certification — Une salle blanche n'est pas un placard avec un purificateur d'air. C'est une installation technique avec sas d'entrée, contrôle continu de la concentration de particules, surveillance de l'humidité et de la température, certifications annuelles. Le coût d'installation et d'entretien est l'une des raisons pour lesquelles la récupération physique professionnelle est facturée plusieurs centaines à plusieurs milliers d'euros.

9.2 La technique reine : le head swap

Quand une ou plusieurs têtes de lecture sont défaillantes (cas le plus fréquent en récupération HDD), la solution est de transplanter l'ensemble têtes/bras depuis un disque donneur identique. C'est le **head swap**.

Étapes en salle blanche :

1. Identification précise du modèle, de la révision firmware et du lot de fabrication du disque patient.

2. Sourcing d'un disque donneur strictement identique. Les laboratoires gardent un stock de centaines de modèles, ou les achètent à des grossistes spécialisés.
3. Ouverture des deux disques sous flux laminaire, démontage des ensembles têtes/bras à l'aide d'outils dédiés (clés spécifiques, séparateurs de têtes pour éviter qu'elles se touchent quand elles sont hors du disque).
4. Transfert de l'ensemble têtes/bras du donneur vers le patient.
5. Refermeture du patient, remise sous tension.
6. Imagerie immédiate avec un imager hardware (PC-3000, DeepSpar), avant que la nouvelle combinaison ne s'use à son tour.

Le donneur doit être identique jusqu'à la révision firmware. Un donneur de modèle apparemment identique mais d'une autre révision de firmware peut donner un disque qui tourne mais ne lit rien : la calibration têtes/firmware est figée au fabricant.

9.3 PCB swap et reprogrammation ROM

Si la panne est sur le circuit imprimé extérieur (surtension, TVS grillé), on peut remplacer le PCB par celui d'un donneur. Mais attention : sur la plupart des disques modernes, des paramètres de calibration spécifiques au support physique (carte des secteurs défectueux, paramètres têtes, etc.) sont stockés dans une ROM sur le PCB. Sans transfert de cette ROM (par dessoudage et resoudage, ou via PC-3000), le disque patient avec un PCB de donneur produira au mieux des données illisibles.

PC-3000 permet aussi de reprogrammer directement la ROM via une connexion COM/UART aux points de test du PCB. C'est rapide et non destructif.

9.4 Cas particuliers

9.4.1 Disques SMR

Les disques en SMR (voir 1.3) posent des défis spécifiques. Quand le firmware ou la zone de translation est corrompu, on ne peut pas simplement lire les plateaux : il faut reconstituer le mapping LBA → emplacement physique en tenant compte du cache et des bandes. PC-3000 a publié des modules SMR dédiés à partir de 2020. Le taux de succès reste inférieur à celui des CMR équivalents.

9.4.2 Stiction

Quand les têtes restent collées au plateau au démarrage, on peut tenter de « décoller » manuellement dans la salle blanche en faisant tourner les plateaux à la main pendant qu'on applique brièvement l'alimentation. Très délicat — on risque d'arracher la couche magnétique sous les têtes.

9.4.3 Plateaux rayés

Si la rayure est superficielle et localisée, on peut souvent récupérer ce qui est en dehors de la zone rayée (avec des secteurs perdus). Si la rayure est profonde ou étendue, c'est terminal — la couche magnétique sous la rayure est arrachée, les données qui y étaient ne sont plus.

Certains laboratoires extrêmement spécialisés pratiquent le **platter swap** : transfert des plateaux d'un disque patient vers un boîtier mécanique de donneur. C'est techniquement le plus délicat (déplacer un plateau sans désynchroniser sa position relative aux têtes est presque impossible) et le taux de succès est faible. À réserver aux cas extrêmes avec enjeu très fort.

9.5 Pourquoi ne pas tenter chez soi

Le calcul est simple : la poussière atmosphérique typique contient des particules de 5 à 50 microns, soit mille fois plus que l'espace entre la tête et le plateau. Une seule particule qui se glisse sur un plateau quand vous ouvrez le boîtier sur votre bureau, et la première rotation après remontage rayera irrémédiablement la surface.

On voit régulièrement, dans les forums de récupération, des messages d'utilisateurs qui ont « voulu essayer » d'ouvrir leur disque dur. Les laboratoires qui reçoivent ces cas dans un second temps constatent presque toujours des dégâts irréversibles. Sur ce type de panne, il n'y a pas de « voir ce que ça donne » — soit on a une salle blanche, soit on n'ouvre pas.

PARTIE III — MÉTHODES

Chapitre 10

Intervention physique sur SSD

10.1 Pourquoi c'est plus dur que sur HDD

Sur un HDD, la donnée est magnétique et persistante. Si on arrive à faire tourner les plateaux et à les lire (par head swap, PCB swap, etc.), on a accès aux bits tels qu'ils ont été écrits.

Sur un SSD, la donnée est :

- Électrique (charge piégée dans des cellules), donc potentiellement volatile dans le temps si elle n'est pas rafraîchie.
- Embrouillée (*scrambled*) par le contrôleur pour équilibrer les charges électriques — il faut savoir désembrouiller.
- Codée par un ECC propre au contrôleur — il faut savoir décoder.
- Mappée logiquement par une **Flash Translation Layer** (FTL) que seul le contrôleur d'origine connaît parfaitement.
- Souvent chiffrée matériellement (SED, TCG Opal) par une clé que seul le contrôleur d'origine peut déverrouiller.

Trois techniques d'intervention, par ordre de préférence (la moins destructive d'abord) :

10.2 JTAG / ISP : non destructif

Les contrôleurs SSD modernes exposent souvent des points de test correspondant à un protocole de débogage interne : **JTAG** (Joint Test Action Group) ou **ISP** (In-System Programming). En soudant temporairement des fils fins sur ces points, et en connectant à un programmeur dédié, on peut :

- Lire le firmware du contrôleur.
- Injecter un *loader* de récupération qui court-circuite le firmware mort et accède directement à la NAND via les capacités du contrôleur.
- Sur certaines configurations, faire parler le contrôleur comme s'il fonctionnait normalement, et imager via SATA ou NVMe.

Avantages : la NAND est lue par son contrôleur d'origine, donc désembrouillage et ECC sont gérés automatiquement, et le **chiffrement matériel reste déchiffré** si la clé est présente. Le support physique n'est pas détruit.

Inconvénients : nécessite des outils spécialisés (PC-3000 Flash avec adaptateurs JTAG, ou solutions tierces comme RTPro, Medusa Pro), de bonnes compétences en micro-soudure de précision, et la connaissance des points de test pour chaque famille de contrôleurs. PC-3000 maintient une base de schémas par contrôleur, accessible aux labos sous contrat.

10.3 Chip-off : la technique de dernier recours

Si JTAG échoue ou n'est pas applicable (contrôleur mort, points de test absents), on désoude physiquement les puces NAND pour les lire indépendamment du contrôleur.

Étape 1 — Désoudage

Les puces NAND sont presque toujours en boîtier **BGA** (Ball Grid Array) — des centaines de billes de soudure sous le composant, invisibles depuis le dessus. Le désoudage nécessite une station de rework à air chaud (ou infrarouge) avec un profil thermique précis : préchauffe progressive du PCB, montée vers les températures de fusion de la soudure (typiquement 220 à 250 °C selon l'alliage), maintien court, et extraction à la pince. Un profil mal calibré endommage la puce avant le désoudage. Un fer à souder simple ne suffit jamais — il concentre la chaleur sur un point et craque la puce.

Étape 2 — Lecture des puces

Les billes de soudure sont nettoyées (les laboratoires utilisent souvent un re-balling pour les remplacer proprement). La puce est ensuite placée sur un **socket** adapté à son format exact (BGA152, BGA132, BGA100, etc., ou TSOP pour les modèles anciens). Le socket est branché sur un programmeur NAND : PC-3000 Flash (ACE Lab), Flash Extractor (Soft-Center), DeepSpar NAND Reader, ou solutions de niche.

Le programmeur lit les pages brutes — typiquement 4, 8 ou 16 Ko de données utilisateur par page, plus une zone **OOB** (Out-Of-Band) de quelques centaines d'octets qui contient les métadonnées et l'ECC.

Les cellules NAND modernes (TLC et QLC) ont une marge entre niveaux de tension étroite. Une lecture unique peut produire des erreurs aléatoires (*bit flips*). Les laboratoires effectuent souvent 3 à 10 lectures successives et fusionnent les résultats par vote majoritaire avant de passer à l'étape suivante.

Étape 3 — Reconstruction logique

C'est l'étape la plus complexe. La puce contient bien des données, mais elles ne sont pas dans un ordre exploitable :

- Le contrôleur appliquait un **scrambler** qui xor-ait les données avec une séquence pseudo-aléatoire, pour équilibrer les charges. Il faut retrouver et désappliquer cet algorithme.
- L'**ECC** protégeait chaque page. Il faut savoir lequel (BCH, Reed-Solomon, LDPC selon les modèles), avec quels paramètres, et le décoder pour corriger les bit-flips résiduels.
- La **FTL** (Flash Translation Layer) mappait les LBA logiques aux pages physiques, avec wear leveling. La même page physique peut contenir plusieurs versions successives d'un même LBA — il faut savoir laquelle est valide.
- Sur SSD multi-puces, les LBA étaient **striped** entre les puces. Il faut reconstituer la séquence.

Aucune de ces étapes n'est documentée par les fabricants de contrôleurs (Phison, Silicon Motion, Samsung, Marvell, Micron). Les laboratoires utilisent des bases de **profils** par contrôleur et firmware, construites par ingénierie inverse et entretenues à grands frais. PC-3000 Flash est la plateforme dominante pour ce travail ; elle intègre des milliers de profils mis à jour à chaque mise à jour logicielle. Sans accès à ces profils, le chip-off est réalisable mais le résultat est presque toujours inexploitable.

10.4 Le mur du chiffrement matériel

Si le SSD utilise un **SED** (Self-Encrypting Drive, norme TCG Opal 2.0), **tout le contenu de la NAND est chiffré** à la volée par le contrôleur, avec une clé AES stockée dans le contrôleur et déverrouillée à chaque démarrage par le mot de passe ATA ou la *pre-boot authentication*. Le chip-off donne du contenu chiffré ; sans la clé maître, c'est définitivement inutilisable.

Cas particulier : BitLocker sur Windows utilise par défaut AES logiciel *sauf* sur SSD compatible TCG Opal où, depuis Windows 10 1903, BitLocker s'appuie sur le chiffrement matériel par défaut (l'option « eDrive »). C'est un débat de sécurité — plusieurs SSD ont eu des implémentations Opal bogguées qui rendaient le chiffrement trivial à contourner, ce qui a poussé Microsoft à proposer de revenir au logiciel par défaut. Mais pour la récupération, l'effet est le même : sans la clé, aucune chance.

Attention — Si vous avez un SSD chiffré et que vous ne disposez plus de la clé (mot de passe oublié, TPM réinitialisé, machine détruite), aucun laboratoire au monde ne récupérera vos données. AES-256 correctement implémenté est mathématiquement indéchiffrable avec les moyens actuels. Cette absence d'option est une fonctionnalité de sécurité, pas un bug. Conséquence pratique : si vos données sont vraiment critiques, stockez vos clés de récupération hors de la machine — dans un gestionnaire de mots de passe, sur papier dans un coffre.

10.5 Cas eMMC et UFS

Les supports embarqués des smartphones et tablettes (eMMC, UFS) sont des variantes de NAND avec contrôleur intégré dans le même boîtier (architecture **monolithique**). Le désoudage est encore plus difficile : la puce est plus petite, et toute l'électronique de contrôle est dedans — il n'y a pas de NAND séparée à lire indépendamment.

Les laboratoires utilisent souvent l'**ISP** (In-System Programming) sur ces supports, qui consiste à souder temporairement aux points de test d'un téléphone *sans dessouder la puce*. Cela permet de lire le contenu via les capacités du contrôleur — pratique sur iPhone et Android quand le chiffrement n'est pas activé ou que la clé est connue.

Sur iPhone récent (à partir de l'A7/A8) et sur la plupart des Android post-2018, le chiffrement de l'eMMC/UFS est activé par défaut, avec une clé liée au passcode utilisateur et à une *Secure Enclave*. Le chip-off ne donne rien d'exploitable. Voir chapitre 13 pour les options spécifiques au mobile.

PARTIE III — MÉTHODES

Chapitre 11

RAID et stockage avancé

11.1 Rappel des niveaux RAID

Pour mémoire :

Niveau	Principe	Tolérance de panne	Récupération si panne
RAID 0	Striping pur, données réparties sur N disques	Aucune — toute panne perd tout	Reconstruction des stripes nécessaire
RAID 1	Miroir : chaque disque est la copie de l'autre (sur N)	1 disque	Triviale : récupérer depuis le miroir sain
RAID 5	Striping + parité distribuée sur N disques	1 disque	Reconstruction possible avec parité ; un 2e échec = catastrophe
RAID 6	Striping + double parité	2 disques	Tolérance plus large, mais reconstruction lente
RAID 10	Miroir + striping (RAID 1+0)	1 disque par miroir	Triviale tant qu'un miroir reste
RAID-Z (ZFS)	Variante RAID 5/6 avec checksums	1 à 3 disques selon Z1/Z2/Z3	Self-healing si tolérance respectée

11.2 Pourquoi le RAID complique la récupération

Un array RAID n'est pas une simple concaténation de disques : c'est une couche logique qui répartit les blocs selon des paramètres (taille de stripe, ordre des disques, décalage de la parité) qu'il faut connaître précisément pour reconstruire l'array.

Quand l'array tourne, le contrôleur (matériel ou logiciel) fait ce travail. Quand l'array tombe — contrôleur en panne, trop de disques HS, métadonnées RAID corrompues, recombinaison manuelle ratée — il faut reconstruire la logique de l'array à la main, en analysant le contenu brut des disques.

Les outils qui font ce travail (R-Studio Network, UFS Explorer RAID, Reclame) procèdent typiquement ainsi :

1. On image chaque disque membre individuellement.
2. On analyse les premiers méga-octets pour détecter les signatures RAID (en-têtes mdadm sur Linux, structures propriétaires sur les contrôleurs matériels).
3. Si les métadonnées sont absentes ou corrompues, on tente une **détection automatique des paramètres** en cherchant des structures FS reconnaissables (en-tête NTFS, superblock ext4) et en testant différentes hypothèses sur la taille de stripe, l'ordre des disques, le sens de rotation de la parité.
4. Une fois les paramètres trouvés, l'array est reconstruit **virtuellement** dans l'outil (pas physiquement) et présenté comme un volume sur lequel on lance l'analyse FS normale.

11.3 La règle d'or : imager d'abord, reconstruire ensuite

L'erreur la plus fréquente : tenter de reconstruire l'array *sur place*, en remettant les disques dans le contrôleur d'origine et en demandant un rebuild. Si l'array est dégradé ou que les métadonnées sont incohérentes, le rebuild écrit sur les disques restants et peut détruire des stripes encore récupérables.

La procédure professionnelle est invariable :

1. Étiqueter physiquement chaque disque membre (position, numéro de série, état observé).
2. Sortir les disques de l'array et de la machine.
3. Imager chaque disque *individuellement* avec ddrescue ou un imager hardware.
4. Travailler exclusivement sur les images, jamais sur les disques d'origine.
5. Reconstruire l'array virtuellement dans l'outil.

11.4 Le cas particulier du RAID 5 dégradé

RAID 5 tolère une panne de disque. Le danger : si un disque est HS et que pendant le rebuild un second disque tombe, **l'array est perdu**. Et statistiquement, c'est exactement ce qui arrive de plus en plus souvent sur les arrays modernes :

- Les capacités de disques augmentent (10, 14, 20, 26 To) mais les taux d'erreur de lecture irrécupérable (URE) restent du même ordre ($\sim 10^{-14}$ bits sur les disques consumer). Sur un rebuild de 20 To, on a une probabilité non négligeable de rencontrer une erreur de lecture quelque part.
- Le rebuild stresse fortement les disques restants (lectures intensives sur plusieurs heures à jours). C'est précisément la situation où un disque marginal va lâcher.

Conséquence : pour les arrays modernes au-delà de quelques To, RAID 5 n'est plus considéré comme suffisant par les architectes stockage. RAID 6 (double parité), RAID 10 ou RAID-Z2/Z3 sont les recommandations courantes.

11.5 RAID matériel vs logiciel

RAID matériel : un contrôleur dédié (PERC Dell, MegaRAID Broadcom/LSI, SmartArray HPE) gère l'array. Les métadonnées RAID sont stockées sur les disques au format propriétaire du contrôleur. Récupération plus complexe si le contrôleur tombe : il faut trouver un contrôleur compatible (même modèle, même firmware idéalement) ou reconstituer l'array à la main.

RAID logiciel : géré par l'OS. Sous Linux, mdadm stocke les métadonnées au format mdraid standard, documenté et récupérable. Sous Windows, Storage Spaces utilise un format propre mais bien analysé. Sous macOS, AppleRAID (limité à RAID 0 et 1) est simple.

Pour une infrastructure neuve, le RAID logiciel est aujourd'hui souvent préférable : récupération plus aisée, indépendance matérielle, pas de dépendance à un contrôleur qu'on ne trouvera peut-être plus dans cinq ans.

11.6 NAS et arrays propriétaires

Les NAS grand public (Synology, QNAP, Asustor) implémentent leurs propres variantes :

- **Synology Hybrid RAID (SHR)** : variante de RAID 5/6 qui permet de mélanger disques de tailles différentes. S'appuie sur mdadm + LVM + Btrfs ou ext4.
- **QNAP RAID** : standard mdadm + ext4 ou ZFS sur certains modèles.
- **TrueNAS** : ZFS natif, avec RAID-Z1/Z2/Z3 selon configuration.

Bonne nouvelle : ces formats sont tous bien supportés par R-Studio Technician, UFS Explorer Professional/RAID Recovery, et Reclame. La récupération depuis un NAS HS suit la même logique : extraire les disques, imager, monter sur une machine Linux ou dans l'outil de récupération.

Vérifier les snapshots d'abord — Sur les NAS Synology et QNAP, la plupart des configurations modernes incluent des snapshots Btrfs automatiques. Avant toute opération de récupération complexe, vérifiez si un snapshot ne contient pas déjà ce qu'on cherche : c'est presque toujours plus rapide.

Partie IV

Cas spéciaux

Trois domaines où les techniques générales des chapitres précédents s'adaptent ou se heurtent à des contraintes spécifiques : le chiffrement (qui peut transformer un cas trivial en impossible), les supports mobiles (smartphones, tablettes), et le contexte judiciaire (où la procédure compte autant que la technique).

PARTIE IV — CAS SPÉCIAUX

Chapitre 12

Chiffrement et récupération

12.1 Une bascule de paradigme

Quand un support n'est pas chiffré, la donnée est lisible par quiconque accède au stockage physique. Quand un support est chiffré, la donnée n'est plus une information ; c'est une suite de bits indistinguable de bruit aléatoire. La récupération devient un problème de cryptanalyse, ce qui veut dire en pratique : **impossible sans la clé**.

C'est une bonne chose pour la sécurité ; c'est un mur infranchissable pour la récupération. Toute la question devient : peut-on retrouver la clé ?

12.2 BitLocker (Windows)

BitLocker chiffre les volumes Windows avec AES-128 ou AES-256. La clé maître (FVEK, *Full Volume Encryption Key*) est protégée par un ou plusieurs **protecteurs** :

- TPM (Trusted Platform Module) : la clé est stockée dans le module sécurisé de la carte mère, et libérée au démarrage si l'état du système est conforme.
- Mot de passe utilisateur.
- Clé de récupération à 48 chiffres (sauvegardée dans le compte Microsoft / Azure AD de l'utilisateur, ou imprimée sur papier).
- Clé USB.
- Carte à puce.

Pour récupérer un volume BitLocker, il faut au moins l'un de ces protecteurs. Bonne piste systématique : la clé de récupération sauvegardée dans le compte Microsoft. À récupérer via account.microsoft.com (compte personnel) ou via l'administrateur AD/Azure (entreprise).

Sans aucun protecteur, AES-128 ou AES-256 correctement implémenté résiste à toutes les attaques connues. Renoncer.

12.3 FileVault (macOS)

FileVault 2 chiffre l'intégralité du volume APFS avec AES-XTS. Sur les Mac avec puce Apple Silicon (M1, M2, M3, M4) le chiffrement est activé **par défaut**, géré par la Secure Enclave, et inhérent à l'architecture matérielle — il ne se désactive pas vraiment.

Voies de récupération possibles :

- Mot de passe utilisateur (le déverrouille au login).
- Clé de récupération à 24 caractères (générée à l'activation de FileVault et que l'utilisateur a normalement notée ou stockée dans iCloud).
- Pour les Mac d'entreprise, clé institutionnelle configurée par le service informatique avant déploiement.

Sans aucune de ces voies, le chip-off n'a aucun intérêt : même en lisant parfaitement les puces NAND, on n'obtient que du chiffré.

12.4 LUKS (Linux)

LUKS (*Linux Unified Key Setup*) est le standard de chiffrement de volume sous Linux. Implémenté par cryptsetup. Le volume LUKS contient une **en-tête** (LUKS header) avec jusqu'à 8 slots de passphrase, et un **master key** chiffré par chaque passphrase.

Récupération :

- Si l'utilisateur connaît la passphrase d'un slot, tout va bien : `cryptsetup open /dev/sdaX nom`.
- Si l'en-tête est corrompue mais qu'on a une sauvegarde (faite avec `cryptsetup luksHeaderBackup`), on peut la restaurer.
- Sans rien : le master key est protégé par Argon2 ou PBKDF2 avec un facteur de coût élevé. Le brute force sur passphrase moderne (12+ caractères mélangés) est impraticable.

Conseil de prévention — Pour les utilisateurs LUKS : sauvegardez votre LUKS header dès le départ et conservez-la hors du disque ! Une simple corruption de l'en-tête (quelques secteurs au début du périphérique) rend le volume définitivement illisible si on n'a pas la sauvegarde.

12.5 Self-Encrypting Drives (SED, TCG Opal)

Le chiffrement matériel intégré au contrôleur SSD est de plus en plus la norme. Il est :

- Toujours actif (la donnée est chiffrée même quand aucun mot de passe n'est défini — le contrôleur « déchiffre » avec une clé par défaut).
- Activable côté utilisateur via BIOS/UEFI (mot de passe ATA) ou logiciel TCG Opal Manager.
- Effaçable instantanément en générant une nouvelle clé maître — c'est la fameuse fonction *crypto erase* qui rend la mise au rebut des SSD sécurisée en quelques secondes.

Pour la récupération : sans mot de passe (s'il a été défini), rien n'est faisable. Avec mot de passe, on déverrouille via ATA security feature set ou commande Opal. Plusieurs SSD ont eu des implémentations Opal défaillantes dans le passé (Crucial MX100/MX200, Samsung 840 EVO/850 EVO d'avant firmware EMT02B6Q) qui permettaient de contourner — mais compter là-dessus est jouer au loto.

12.6 Le ransomware : ce qui est réellement possible

Quand un poste ou un serveur est touché par un ransomware, les fichiers utilisateurs sont chiffrés avec une clé (typiquement AES) elle-même chiffrée par la clé publique de l'attaquant. Sans la clé privée correspondante — détenue par l'attaquant — le déchiffrement est mathématiquement impossible.

Voies à examiner systématiquement avant de céder au désespoir :

1. **Existe-t-il un decryptor public ?** Le projet *No More Ransom* (nomoreransom.org), porté par Europol, publie des outils gratuits pour les variantes dont les clés ont été saisies ou les failles cryptographiques découvertes. Plus de 200 outils disponibles en 2026.
2. **La machine est-elle encore allumée ?** Si l'attaque est récente et que la machine n'a pas été redémarrée, la clé AES en clair peut être encore en mémoire vive. Une analyse RAM (dumpit,

Volatility) peut la récupérer.

3. **Existe-t-il un Shadow Copy ?** Sur Windows, vssadmin peut révéler des Volume Shadow Copies que le ransomware n'a pas réussi à supprimer.
4. **Les sauvegardes sont-elles intactes ?** Question primordiale. Le ransomware moderne cherche et supprime activement les sauvegardes. Si elles sont immuables ou air-gappées (voir chapitre 17), on est sauvé.
5. **Faut-il payer ?** Question complexe et déconseillée par les autorités, mais à laquelle 36 % des victimes en 2024 ont encore répondu oui. Cela n'offre *aucune garantie* de déchiffrement.

Attention — Conserver les fichiers chiffrés même si on ne peut pas les déchiffrer aujourd'hui. Les clés de ransomwares historiques sont régulièrement saisies par les forces de l'ordre (LockBit, Hive, REvil, etc.) et publiées des mois ou années après l'attaque. Stockez les fichiers chiffrés sur un disque hors-ligne ; un jour, vous pourrez peut-être les déchiffrer.

12.7 Étude de cas — Maersk et NotPetya (juin 2017)

■ Maersk / NotPetya : la sauvegarde sauvée par une panne de courant

Le 27 juin 2017, le géant maritime Maersk est touché par **NotPetya**, malware destructeur déguisé en ransomware, propagé via une mise à jour piégée du logiciel comptable ukrainien M.E.Doc. En 7 minutes, le malware se propage à travers tout le réseau Maersk : 45 000 à 49 000 postes, 4 000 serveurs détruits, dont la totalité des ~150 **contrôleurs de domaine Active Directory**. NotPetya n'est pas réversible (le déchiffrement est impossible par construction) : les machines sont mortes.

Maersk a des sauvegardes des serveurs individuels (entre 3 et 7 jours d'âge selon les cas), mais aucune sauvegarde des contrôleurs de domaine — l'architecture supposait que les 150 contrôleurs se sauvegardaient mutuellement par réplication. Or ils ont tous été détruits simultanément. Sans AD, rien ne peut être restauré.

Salvation : un contrôleur de domaine au Ghana était **hors ligne** au moment de l'attaque, à cause d'une panne de courant locale. Il avait survécu. Maersk l'a fait acheminer physiquement (le réseau étant détruit) à Londres où le centre de récupération avait été monté. Ce contrôleur a servi de base pour reconstruire la totalité de l'infrastructure.

Bilan : 10 jours de paralysie totale, perte estimée 250-300 millions de dollars pour Maersk (estimation interne, considérée comme prudente). Au global, NotPetya a coûté environ 10 milliards de dollars à travers Merck, FedEx/TNT, Mondelez, Saint-Gobain et autres. Sources : témoignages CISO Maersk au Gartner Risk Summit 2019 ; Wired, *The Untold Story of NotPetya*, 2018 ; Control Engineering, *Throwback Attack*, 2025.

Leçon : les sauvegardes en ligne et synchrones entre elles ne protègent pas contre une attaque qui les détruit toutes en même temps. La survie est venue d'un hasard (une panne de courant qui a déconnecté un serveur juste à temps). Une sauvegarde air-gappée ou immuable aurait évité le hasard.

PARTIE IV — CAS SPÉCIAUX

Chapitre 13

Supports mobiles (Android, iOS)

13.1 Le contexte

Un smartphone moderne contient typiquement plus de données personnelles qu'un ordinateur de bureau : photos, messages, historique de localisation, contacts, applications professionnelles, comptes synchronisés. Mais c'est aussi un des supports les plus difficiles à récupérer, pour trois raisons :

- Chiffrement par défaut activé sur tous les appareils récents (iPhone depuis l'iPhone 3GS en 2010 ; Android depuis Android 6 Marshmallow en 2015 en pratique).
- Stockage de type eMMC ou UFS, monolithique (NAND et contrôleur dans une même puce), très difficile à désouder et à lire indépendamment.
- Liaison forte entre l'appareil et un compte cloud (Apple ID, Google account) qui rend toute manipulation post-mortem complexe.

13.2 iPhone et iOS

Architecture clé : la **Secure Enclave** (présente depuis l'iPhone 5s, 2013), un coprocesseur cryptographique séparé qui stocke les clés et applique une politique stricte sur le déchiffrement. Toute donnée utilisateur sur le stockage NAND est chiffrée par une clé liée à la fois au passcode utilisateur et à un UID matériel inscrit dans la Secure Enclave.

Conséquences :

- Le chip-off d'un iPhone moderne donne du chiffré ininterprétable. La Secure Enclave ne donne sa clé à personne sans le passcode.
- Les outils forensiques iPhone professionnels (Cellebrite UFED, GrayKey de Grayshift, Magnet GrayKey) exploitent des failles non publiques de certaines versions d'iOS pour contourner la Secure Enclave. Ils fonctionnent uniquement sur les modèles et versions vulnérables, et les fenêtres se ferment à chaque mise à jour.
- Pour le particulier : les options sont la **sauvegarde iTunes/Finder** (sur un Mac de confiance) ou la **sauvegarde iCloud**. Aucune récupération depuis l'iPhone lui-même sans le passcode.

13.3 Android

Plus hétérogène : Android tourne sur des centaines de modèles, avec des implémentations matérielles et logicielles variables. Globalement :

- Chiffrement par défaut depuis Android 6, basé sur le passcode et un keystore matériel (**TEE**, Trusted Execution Environment).
- Android 10+ utilise le **chiffrement par fichier** (file-based encryption) qui permet à certaines fonctions (alarmes, accessibilité) de fonctionner avant déverrouillage, tout en chiffrant les données utilisateur.

Outils de récupération mobile :

- **Cellebrite UFED, MSAB XRY, Oxygen Forensic Detective, Magnet AXIOM** : suites professionnelles pour les forces de l'ordre et entreprises (coûts élevés, licences contrôlées).
- **ADB** (Android Debug Bridge) : pour récupérer depuis un téléphone vivant et déverrouillé, avec le débogage USB activé. Permet de copier les données accessibles côté utilisateur.
- Mode **Download / EDL** sur certaines puces Qualcomm : permet aux laboratoires un accès bas-niveau, mais l'EDL est désormais signé par le fabricant et donc restreint.

Pour un Android verrouillé qu'on a perdu, la voie la plus praticable est la **sauvegarde Google** : photos sur Google Photos, contacts/agenda/Drive sur le compte. Si la synchronisation était activée, beaucoup de données sont récupérables sans accès au téléphone.

13.4 Le rôle des sauvegardes cloud

Pour les supports mobiles, c'est presque toujours la voie la plus productive. Récupérer depuis :

- **iCloud** : Photos, contacts, mail, documents iCloud Drive, sauvegardes complètes iOS (si activées).
- **Google** : Photos, contacts, agenda, Drive, sauvegardes Android (Google One).
- **Comptes tiers** : WhatsApp (sauvegardes Google Drive / iCloud), Telegram (cloud natif), Signal (export local), etc.

Bien dire à l'utilisateur que ces sauvegardes existent et comment y accéder. La majorité des particuliers en perte de données mobile ignore ce qu'ils ont sauvegardé automatiquement.

PARTIE IV — CAS SPÉCIAUX

Chapitre 14

Forensique judiciaire

14.1 La différence fondamentale

La récupération de données classique a un objectif simple : récupérer ce qui peut l'être. La forensique judiciaire en ajoute un second : **produire un résultat recevable devant un tribunal**. La procédure devient aussi importante que le résultat technique.

Concrètement, cela implique :

- Documentation continue de chaque manipulation (*chain of custody*).
- Vérification d'intégrité par hash à chaque étape.
- Usage de write blockers pour garantir que le support original n'a pas été modifié.
- Reproductibilité : un autre expert doit pouvoir refaire la même analyse et obtenir les mêmes résultats.
- Outils reconnus et validés (validés scientifiquement, utilisés par les pairs, documentés).

14.2 ISO 27037 : la norme de référence

La norme **ISO/IEC 27037:2012** (« Lignes directrices pour l'identification, la collecte, l'acquisition et la préservation de preuves numériques ») définit le cadre international. Elle distingue quatre phases :

1. **Identification** : localiser et reconnaître les supports potentiellement porteurs de preuves.
2. **Collecte** : prendre possession physique de manière documentée.
3. **Acquisition** : créer une copie forensique avec vérification d'intégrité.
4. **Préservation** : maintenir l'intégrité dans le temps, avec chaîne de custody documentée.

En France, l'expert judiciaire en informatique opère dans le cadre du Code de procédure pénale (articles 60, 156 et suivants pour les expertises) et doit être inscrit sur la liste des experts près une cour d'appel. Au pénal, c'est souvent la PJ (sous-direction de la lutte contre la cybercriminalité) qui effectue les saisies, avec l'appui d'experts.

14.3 Outils forensiques de référence

Les suites forensiques professionnelles couvrent toute la chaîne acquisition + analyse + reporting. Les références internationales en 2026 :

- **EnCase Forensic** (OpenText) : suite historique (depuis Guidance Software, 1997), très utilisée par les forces de l'ordre US.
- **FTK** (AccessData / Exterro) : concurrent direct d'EnCase, FTK Imager seul est gratuit.
- **X-Ways Forensics** (X-Ways Software, Allemagne) : référence européenne, plus léger et plus rapide qu'EnCase/FTK.
- **Magnet AXIOM** : focus sur les artefacts cloud, mobile et navigateur.
- **Belkasoft X** : très bon sur les applications mobiles et messagerie.

- **The Sleuth Kit + Autopsy** : la suite open source, gratuite. Largement suffisante pour beaucoup de cas, et utilisée dans la formation forensique académique.

14.4 Workflow forensique type

1. Documenter l'état initial du support (photos, numéros de série, état physique). Notation manuelle ou logiciel de gestion de scellés.
2. Connecter le support à la station d'analyse via un write blocker matériel (Tableau, WiebeTech).
3. Créer une image forensique (E01 ou DD brut) avec calcul automatique de hash (MD5 + SHA-1 ou SHA-256).
4. Vérifier que le hash de l'image correspond à un hash calculé directement sur la source. Sceller la source.
5. Travailler sur l'image. Toute analyse, tout export (fichiers récupérés, timeline, artefacts) est consigné dans le rapport.
6. Le rapport final inclut : description du support, procédure d'acquisition, hashes, outils utilisés avec versions, méthodologie d'analyse, conclusions, liste des artefacts produits.

14.5 Artefacts à analyser sur Windows

Au-delà des fichiers utilisateurs, un système Windows contient des dizaines d'artefacts précieux pour reconstituer l'activité :

- **Registre** (NTUSER.DAT, SOFTWARE, SYSTEM, SAM, SECURITY) : configuration, comptes, USB connectés, programmes exécutés (UserAssist, ShellBags, MUICache).
- **Prefetch** : trace des programmes exécutés récemment, avec compteur.
- **ShellBags** : dossiers visualisés dans l'Explorateur.
- **RecycleBin** : fichiers supprimés via la corbeille (avec métadonnées originales).
- **\$LogFile et \$UsnJrnl** (chapitre 7) : journal des modifications NTFS.
- **Event Logs** (.evtx) : événements système, sécurité, applications.
- **Browsers** : historique, cookies, cache, téléchargements (Chrome dans %LocalAppData%\Google\Chrome\User Data\, Firefox dans %AppData%\Mozilla\Firefox\Profiles\).

14.6 Précautions et erreurs courantes

Attention — Ne *jamais* démarrer la machine cible sur son OS d'origine. Tout démarrage modifie des centaines de fichiers (timestamps, logs, registre), ce qui peut suffire à invalider la preuve. Toujours retirer le disque ou démarrer sur un live forensique en lecture seule.

Autres erreurs fréquentes :

- Oublier de calculer le hash avant analyse — rend impossible de prouver que l'image n'a pas été modifiée.
- Travailler sur le support source au lieu de l'image — toute analyse non lecture-seule modifie potentiellement la preuve.

- Utiliser des outils non documentés ou maison sans valider leur fonctionnement par des tests reproductibles.
- Omettre de tracer une étape du processus dans le rapport — une trace incomplète peut donner prise à la défense pour remettre en cause la totalité de l'analyse.

Partie V**Pratique**

Deux chapitres pour passer de la théorie aux choix concrets : quels outils utiliser dans quelle situation, et quelles erreurs éviter, illustrées par des incidents publics réels.

PARTIE V — PRATIQUE

Chapitre 15

Outils 2026 : panorama réaliste

15.1 Méthode de présentation

Ce chapitre liste les outils de récupération sans donner de « note sur 5 » ou de « taux de récupération » chiffré. Ces classements qu'on voit sur internet sont presque toujours :

- Issus de sites affiliés des éditeurs concernés.
- Basés sur des tests qui ne sont pas reproductibles.
- Souvent obsolètes ou copiés-collés d'année en année.

Ce qui suit est une description fonctionnelle et un positionnement honnête, à charge pour le lecteur d'essayer la version gratuite/démo des outils qui paraissent adaptés à son cas.

15.2 Outils open source

ddrescue (GNU)

L'outil incontournable pour l'imagerie. Multiplate-forme via WSL/Linux/Mac. Voir chapitre 6 pour le workflow détaillé.

TestDisk (CGSecurity)

Réparation de tables de partition, récupération de partitions effacées. Fonctionne sur Windows, macOS et Linux. Interface ncurses austère mais efficace. Pour beaucoup de cas logiques courants, c'est tout ce dont on a besoin.

PhotoRec (CGSecurity)

Carving signature-based. Plus de 480 formats reconnus. Multiplate-forme. Pour le carving sans prétention sémantique, c'est la référence open source.

The Sleuth Kit + Autopsy

Suite forensique complète. TSK fournit les outils en ligne de commande (fls, icat, blkls, tsk_recover...), Autopsy une interface web pour l'analyse. Très utilisée en formation et dans certaines unités de police.

extundelete, ext4magic, debugfs

La triade Linux pour ext4. extundelete et ext4magic exploitent le journal ; debugfs donne l'accès interactif au FS.

Outils d'Eric Zimmerman

Suite d'utilitaires forensiques gratuits pour Windows : **MFTECmd** (parse \$MFT), **RECmd** (registre), **LECmd** (LNK), **JLECmd** (Jump Lists), **PECmd** (Prefetch), **Timeline Explorer...** Devenus références du DFIR Windows. Téléchargeables sur ericzimmerman.github.io.

15.3 Outils commerciaux pour particuliers et PME

Disk Drill (CleverFiles)

Windows et macOS. Interface très accessible pour les non-techniciens. Plusieurs algorithmes de scan, support de nombreux formats. Version gratuite limitée à 500 Mo de récupération (Windows).

EaseUS Data Recovery Wizard

Interface très propre, support multi-FS, version gratuite jusqu'à 2 Go. Largement utilisé dans le grand public.

Recuva (CCleaner)

Gratuit, simple. Adapté aux suppressions accidentelles récentes sur Windows. Limité face à des cas complexes.

Stellar Data Recovery

Gamme complète, du grand public au laboratoire. Bonne réputation sur la récupération de fichiers Office et multimédia. Version pro chère.

15.4 Outils professionnels

R-Studio (R-Tools Technology)

Référence pour les techniciens IT et petits laboratoires. Très bon sur NTFS, ext4, APFS, RAID. Inclut un éditeur hex et un module RAID complet. Disponible en plusieurs éditions (consumer, technician, network).

UFS Explorer (SysDev Laboratories)

Excellent sur les formats avancés (APFS, ZFS, Btrfs, NAS Synology/QNAP, RAID complexes). L'édition Professional est une des plus complètes du marché. Très utilisée dans les laboratoires européens.

ReclaiMe et ReclaiMe Pro

Spécialisés dans la reconstruction RAID et l'analyse de configurations propriétaires (Synology Hybrid RAID, ZFS, Microsoft Storage Spaces). Édition Pro très orientée labo.

15.5 Outils forensiques judiciaires

Suites professionnelles présentées au chapitre 14 : EnCase, FTK, X-Ways, Magnet AXIOM, Belkasoft X, Cellebrite UFED, Oxygen Forensic. Réservées aux institutions et entreprises spécialisées (coûts annuels de licences à 5 chiffres pour les unités complètes).

15.6 Plateformes hardware de laboratoire

Pour la récupération physique professionnelle, deux plateformes dominant :

- **PC-3000** (ACE Lab) : standard de facto, des modules pour HDD, SSD, Flash, RAID, mobile. Base de connaissances par contrôleur et firmware tenue à jour. Sous contrat avec engagement annuel.
- **DeepSpar / Atola** : alternatives plus ciblées (imager hardware, head bypass). Coexistent avec PC-3000 dans beaucoup de laboratoires.

Le coût d'équipement complet d'un laboratoire (PC-3000 HDD + Flash + Express + adaptateurs + outils micro-soudure + salle blanche) se chiffre en dizaines de milliers d'euros. C'est la raison principale pour laquelle la récupération physique est facturée à plusieurs centaines d'euros minimum.

15.7 Matrice de décision

Situation	Outils à essayer en premier
Suppression accidentelle récente sur HDD	TestDisk + PhotoRec (gratuit) ou Disk Drill / EaseUS
Suppression sur SSD (mais TRIM-mable)	Tenter Recuva / EaseUS sans grand espoir ; SSD débranché → labo
Volume devenu RAW (NTFS/exFAT corrompu)	TestDisk pour la partition, R-Studio ou UFS Explorer pour le FS
NAS Synology/QNAP HS	Sortir les disques, UFS Explorer Professional ou ReclaiMe Pro
RAID 5 dégradé	Imager d'abord, puis R-Studio Network ou UFS Explorer RAID
Mac sous FileVault, clé connue	Démarrer en target disk mode, copier ; ou R-Studio for Mac
Mac sous FileVault, clé inconnue	Vérifier iCloud (clé de récupération) ; sinon renoncer
HDD qui claque	Éteindre immédiatement → labo (jamais soi-même)
SSD non détecté	Labo (JTAG / chip-off)
Cas judiciaire (preuve à produire)	FTK Imager (gratuit) + Autopsy, ou suite pro (X-Ways, EnCase, AXIOM)
Téléphone Android verrouillé	Sauvegarde Google ; pour le forensique : Cellebrite, MSAB
iPhone verrouillé	Sauvegarde iCloud ; pour le forensique : GrayKey (modèles vulnérables)

PARTIE V — PRATIQUE**Chapitre 16**

Pièges mortels et études de cas

16.1 Les sept erreurs qui tuent les données

1. **Installer un logiciel de récupération sur le support source.** L'installation écrit physiquement sur le support, souvent à des endroits stratégiques (zone libre récemment désallouée — c'est-à-dire exactement là où sont vos fichiers supprimés).
2. **Récupérer les fichiers sur le support source.** Variante de la précédente, et tout aussi catastrophique.
3. **Laisser un SSD sous tension après l'incident.** TRIM et garbage collection continuent leur travail. Sur SSD moderne, chaque minute compte.
4. **Accepter la réparation automatique de Windows.** `chkdsk /f`, `autorepair`, démarrage répété sur un FS corrompu : Windows écrit, déplace, supprime activement des structures pour « réparer ». Sur un FS qu'on veut récupérer, c'est le contraire de ce qu'on veut.
5. **Ouvrir un HDD hors salle blanche.** Voir chapitre 9.
6. **Tenter de dessouder une puce NAND avec un fer à souder.** Sans station de rework et profil thermique calibré, la puce est détruite avant d'être lue.
7. **Conserver les sauvegardes dans le même environnement que la production.** Cas Code Spaces (ci-dessous). Le ransomware moderne cible explicitement les sauvegardes accessibles.

16.2 Étude de cas — Code Spaces (juin 2014)

■ Code Spaces : 12 heures de l'entreprise à la disparition

Code Spaces était une plateforme de code hosting (Subversion et Git) avec environ 7 ans d'historique, basée à Coventry au Royaume-Uni, intégralement hébergée sur AWS.

Le 17 juin 2014, l'entreprise subit d'abord une attaque DDoS, qu'elle dit « courante » et habituellement résolue facilement. Cette fois, l'attaque s'accompagne d'un message extorquant un paiement, laissé directement dans la console EC2 d'AWS. L'attaquant a obtenu l'accès au panneau de contrôle Amazon — par un moyen jamais identifié publiquement (probablement compromission de credentials, sans MFA activé).

Code Spaces refuse de payer et tente de reprendre le contrôle en changeant les mots de passe EC2. Mais l'attaquant avait déjà créé plusieurs comptes de connexion en arrière-plan. Réalisant que Code Spaces essayait de reprendre la main, il lance une suppression méthodique : **EBS snapshots, S3 buckets, AMI, instances EC2, instances de stockage.**

Le point critique : *les sauvegardes de Code Spaces étaient dans le même compte AWS que la production.* Une fois l'accès au panneau obtenu, l'attaquant a pu supprimer simultanément la production et toutes les copies de sauvegarde.

Le 18 juin 2014 — 12 heures après le début de l'attaque — Code Spaces publie un communiqué annonçant la cessation définitive d'activité : « Code Spaces will not be able to operate beyond this point. » Les clients sont invités à exporter ce qui leur reste avant la fermeture finale.

Sources : Threatpost (juin 2014) ; eSecurity Planet (2014) ; InfoWorld *Murder in the Amazon cloud* (2014) ; analyse Wiz *breaches.cloud* (2023).

Leçons : (1) ne jamais stocker les sauvegardes dans le même compte/domaine que la production ; (2) MFA obligatoire sur tout compte de gestion cloud ; (3) principe of least privilege — un seul compte ne doit jamais pouvoir tout effacer ; (4) avoir un plan de réponse à incident testé, pas juste rédigé.

16.3 Quand savoir s'arrêter

Pour les particuliers et les techniciens IT, savoir quand ne pas insister est essentiel. Quatre signes :

- Le support est physiquement détérioré (bruit, non-détection, surchauffe). Pas d'intervention domestique.
- Vous avez déjà tenté plusieurs outils sans résultat — chaque tentative supplémentaire risque d'aggraver.
- Les données ont une valeur supérieure au coût d'une intervention professionnelle. Pour quelques centaines d'euros, on évite parfois la perte définitive.
- Il y a un enjeu judiciaire — toute manipulation amateur risque d'invalidier la preuve.

16.4 Comment choisir un laboratoire

Au-delà des critères du chapitre 5 (salle blanche ISO 5, diagnostic gratuit, paiement au résultat), quelques vérifications complémentaires :

- Demander une visite ou des photos vérifiables de la salle blanche.
- Demander la liste des techniciens et leur expérience (ce ne sont pas des stagiaires qui doivent toucher un disque à 10 000 euros de données).
- Vérifier les avis publics — mais avec discernement : beaucoup sont faux dans les deux sens. Préférer les avis détaillés et anciens.
- Demander à voir un rapport-type pour évaluer le sérieux du livrable.
- Confidentialité contractuelle écrite (NDA) avant toute remise du support.
- Politique de destruction des copies intermédiaires après remise des données et acceptation du client.

Partie VI

Prévention

La meilleure récupération est celle qu'on n'a jamais à faire. Deux chapitres pour fermer le cercle : les stratégies de sauvegarde modernes, et un point honnête sur ce qui reste définitivement hors d'atteinte de la récupération en 2026.

PARTIE VI — PRÉVENTION

Chapitre 17

Stratégies de sauvegarde modernes

17.1 La règle 3-2-1 et son extension

La **règle 3-2-1** a été formulée en 2005 par Peter Krogh, photographe, dans *The DAM Book: Digital Asset Management for Photographers*. Elle se résume à :

- **3** copies de chaque donnée (original + deux sauvegardes).
- **2** supports différents (par exemple : disque interne + disque externe ; ou disque + cloud).
- **1** copie hors site (à l'épreuve des incendies, vols, inondations locales).

Vingt ans plus tard, l'omniprésence du ransomware a poussé Veeam à proposer une extension **3-2-1-1-0** :

- **+1** : une copie supplémentaire qui est *immuable* ou *air-gappée* — qu'un attaquant ayant compromis les comptes ne puisse pas supprimer.
- **+0** : zéro erreur à la restauration, vérifiée par tests réguliers.

Sur l'origine — La règle 3-2-1-1-0 est marketing Veeam, pas un standard ANSSI ou NIST. Cela ne lui enlève rien sur le fond : l'immuabilité et l'air-gap sont devenues incontournables face au ransomware moderne. Les principes sont largement repris par d'autres acteurs (Object First, Wasabi, Backblaze B2 avec Object Lock, Azure Blob immutability, Synology SnapLock).

17.2 Mettre en œuvre l'immuabilité

Plusieurs voies, du plus simple au plus sophistiqué :

- **Object Lock S3** (AWS, Backblaze B2, Wasabi, MinIO). Mode *compliance* ou *governance*. Une fois la donnée écrite, elle ne peut être ni modifiée ni supprimée avant l'expiration de la rétention — même par le propriétaire du bucket.
- **Azure Blob immutability** et Google Cloud Storage retention policies. Mêmes principes que S3 Object Lock.
- **Hardened Linux Repository** (Veeam, mais principe généralisable) : un serveur Linux avec attribut `chattr +i` sur les fichiers de sauvegarde, et SSH désactivé pour root. Faisable maison avec `rsync` et un cron.
- **WORM** (Write Once Read Many) sur bandes ou disques optiques. Solution historique, encore pertinente pour les archives long-terme.
- **Snapshots NAS** immuables : Synology SnapLock, QNAP WORM. À condition de protéger le NAS lui-même (mot de passe administrateur fort, MFA, pas d'accès domaine joint).

17.3 L'air-gap

Un support **air-gappé** est physiquement déconnecté du réseau pendant la majorité du temps.

Variantes :

- **Bande LTO** sortie du robot et stockée dans un coffre. La solution historique, toujours pertinente pour les volumes importants à long terme.
- **Disque USB** qu'on connecte uniquement pour la sauvegarde, puis qu'on range dans un tiroir verrouillé.
- **Rotation de plusieurs disques externes** (par exemple 4 disques, un par semaine du mois) avec un stockage hors site rotatif.

L'air-gap garantit une chose simple : si l'attaquant prend le contrôle du système le mardi à 14 h, il ne peut pas supprimer la sauvegarde du mardi précédent qui dort dans un tiroir éteint.

17.4 La vérification de restauration

Le **0** de 3-2-1-1-0 est sans doute le plus négligé. Beaucoup d'organisations ont des sauvegardes qui n'ont jamais été testées. Le résultat typique : au moment de l'incident, la restauration échoue. Causes fréquentes :

- Corruption silencieuse des fichiers de sauvegarde (bit rot non détecté).
- Sauvegardes incrémentales avec une chaîne cassée à un point quelconque (un seul fichier manquant invalide tout ce qui suit).
- Sauvegardes qui se sont arrêtées silencieusement il y a des mois — l'agent a planté, personne n'a vu.
- Application qui ne peut pas démarrer correctement à partir de la sauvegarde (dépendance non sauvegardée, mauvaise version, base de données en état incohérent).

Bonne pratique : test de restauration **trimestriel** minimum, sur un système isolé. Pour les systèmes critiques, **mensuel**. Documenter chaque test (date, succès/échec, temps de restauration mesuré). C'est la seule manière de savoir que ses sauvegardes valent quelque chose.

17.5 Pour le particulier

Une stratégie pragmatique et abordable :

1. **Disque externe** à 50-100 euros, branché une fois par semaine pour Time Machine (macOS), File History (Windows) ou rsync (Linux).
2. **Cloud personnel** : iCloud, Google One, Dropbox, OneDrive, ou Backblaze Personal Backup (~80 euros/an, sauvegarde tout disque externe inclus).
3. **Pour les fichiers vraiment irremplaçables** (photos, documents administratifs scannés, manuscrits) : une troisième copie sur clé USB stockée chez un proche.
4. **Test annuel** : essayer de restaurer un fichier au hasard à partir de chacune des trois copies. Si l'un échoue, c'est qu'il faut remplacer ou reconfigurer.

17.6 Pour les PME

Niveau de protection minimum recommandé en 2026 :

- Sauvegarde quotidienne automatique des serveurs et postes critiques (Veeam, Acronis, Datto, Synology Active Backup, Rubrik...).
- Au moins une copie en cloud immuable (S3 Object Lock ou équivalent).
- Au moins une copie air-gappée hebdomadaire (bande ou disque rotation).

- MFA obligatoire sur tous les comptes d'administration cloud et sauvegarde.
- Test de restauration mensuel formellement documenté.
- Plan de reprise écrit : qui fait quoi en cas d'incident, dans quel ordre, avec quels contacts.
- Exercice annuel : déclencher un incident fictif, voir si le plan tient.

PARTIE VI — PRÉVENTION

Chapitre 18

Limites actuelles en 2026

18.1 Ce qui est définitivement perdu

Pour ne pas entretenir d'illusions, voici ce qu'aucun laboratoire au monde ne récupère en 2026 :

1. **SSD avec TRIM passé et garbage collection effectuée.** Les cellules NAND ont reçu la tension d'effacement et sont revenues à leur état neutre. Lire la NAND donne des zéros. Aucune technique connue ne récupère.
2. **Données chiffrées par AES-256 sans la clé.** Le déchiffrement par force brute est mathématiquement irréalisable avec les ordinateurs classiques actuels. L'ordinateur quantique de demain pourrait théoriquement casser AES-128 par Grover, mais pas AES-256 ; et ces ordinateurs n'existent pas à l'échelle utile en 2026.
3. **Plateaux HDD avec la couche magnétique arrachée.** L'information était dans le métal. Une fois enlevé, il n'y a plus rien à lire.
4. **Fichiers chiffrés par un ransomware moderne sans la clé et sans erreur d'implémentation.** Mêmes raisons que (2).
5. **RAID 5 avec plus d'un disque HS, RAID 6 avec plus de deux.** La parité ne suffit plus à reconstruire. Sauf récupération individuelle disque par disque (chacun de ces disques peut contenir des stripes utilisables si on les image, mais on n'aura jamais le contenu complet).
6. **Données écrasées par une réécriture complète.** Le mythe de la « rémanence magnétique » (idée que les bits écrasés laisseraient une trace lisible par microscope à force atomique) est techniquement démenti pour les disques modernes. Un seul passage de zéros sur un HDD moderne rend la donnée originale irrécupérable.

18.2 Ce qui est en train de devenir difficile

- **Récupération mobile** : les Secure Enclave iPhone, les TEE Android, le chiffrement par défaut activé partout, ferment progressivement les portes. Les outils forensiques (Cellebrite, GrayKey) exploitent des failles qui se ferment à chaque mise à jour.
- **Récupération SSD classique** : les implémentations TRIM sont devenues fiables, le chiffrement matériel par défaut se généralise. Les cas favorables (TRIM cassé, firmware ancien, ponts USB non-UASP) se raréfient.
- **Récupération cloud** : les politiques de rétention et de suppression côté providers sont de plus en plus rigides. Une suppression définitive sur un service cloud reste définitive.

18.3 Tendances 2026-2030

Quelques évolutions probables :

- Diffusion du **QLC et PLC** sur le SSD grand public : densité plus haute, endurance plus basse, marge entre niveaux de tension plus étroite. Lecture chip-off encore plus difficile.
- Généralisation du **HAMR** sur les HDD haute capacité au-delà de 30 To. Densité de stockage plus élevée, mais pas de changement fondamental pour la récupération.

- **IA dans le carving** et la classification d'artefacts forensiques. Probablement utile sur des cas spécifiques (réassemblage de fragments d'images, identification d'artefacts dans des logs massifs), pas miracle global.
- **Tensions cryptographiques** : si l'ère post-quantique impose le remplacement d'AES par des algorithmes plus jeunes, des erreurs d'implémentation peuvent rouvrir temporairement des fenêtres de récupération.
- **Régulation** : les exigences RGPD/NIS2 obligent à mieux protéger et tracer les supports. Cela aide la prévention plus que la récupération, mais permet d'éviter beaucoup de cas.

18.4 Le message final

Si vous arrivez ici, vous avez compris l'essentiel : la récupération de données est une discipline réelle, techniquement complexe, qui a fait d'énormes progrès mais se heurte à des limites physiques et mathématiques de plus en plus serrées. Les méthodes des chapitres 6 à 11 fonctionnent dans une majorité de cas, mais dépendent presque toujours du temps qui s'écoule entre l'incident et la première bonne décision.

La meilleure stratégie reste, sans surprise, de ne pas avoir à récupérer : prévention, sauvegarde réelle (testée), discipline opérationnelle quand quelque chose tourne mal. Le chapitre 17 est, en pratique, le plus utile du livre.

Bonne lecture, et bonnes sauvegardes.

Annexes

Référence rapide

Trois annexes : commandes de référence pour les outils en ligne de commande les plus utiles, glossaire des termes techniques, et bibliographie complète des sources publiques consultées pour la rédaction de ce manuel.

ANNEXES**Chapitre A****Commandes de référence****A.1 Identification du support**

```
# Linux : lister les disques avec modèle et numéro de série
$ lsblk -o NAME,SIZE,MODEL,SERIAL,TRAN
$ sudo hdparm -I /dev/sdX
$ sudo smartctl -a /dev/sdX

# macOS : informations détaillées
$ diskutil list
$ diskutil info /dev/diskN
$ system_profiler SPSerialATADataType

# Windows (PowerShell)
PS> Get-PhysicalDisk
PS> Get-Disk | Format-List
```

A.2 Imagerie avec ddrescue

```
# Première passe : copier ce qui se lit facilement (sans retry)
$ sudo ddrescue -f -n -d /dev/sdX image.img image.map

# Deuxième passe : retry sur les zones difficiles
$ sudo ddrescue -f -d -r3 /dev/sdX image.img image.map

# Troisième passe : lecture inverse pour les cas sévères
$ sudo ddrescue -f -d -R -r3 /dev/sdX image.img image.map

# Voir l'avancement (depuis un autre terminal)
$ cat image.map | head
$ ddrescue log -t image.map # statistiques
```

A.3 Monter une image en lecture seule

```
# Linux : créer un loop device et monter
$ sudo losetup --read-only --find --show image.img
/dev/loop0
$ sudo blkid /dev/loop0 # voir le type de FS

# Si l'image contient une table de partition :
$ sudo partx --show /dev/loop0 # voir les partitions
$ sudo partx --add /dev/loop0 # créer /dev/loop0p1, p2...

# Monter en lecture seule selon le FS
$ sudo mount -o ro,noexec /dev/loop0p1 /mnt/recup # ext4
$ sudo mount -t ntfs-3g -o ro,norecover /dev/loop0p1 /mnt/recup # NTFS
$ sudo mount -o ro /dev/loop0p1 /mnt/recup # FAT/exFAT

# Démontage et libération
$ sudo umount /mnt/recup
$ sudo partx --delete /dev/loop0
$ sudo losetup -d /dev/loop0
```

A.4 Hash d'intégrité

```
# Calcul
$ sha256sum image.img > image.img.sha256
$ md5sum image.img > image.img.md5

# Vérification ultérieure
$ sha256sum -c image.img.sha256

# Sur de très gros fichiers, calcul parallèle avec b3sum (BLAKE3)
$ b3sum image.img
```

A.5 Analyse NTFS

```
# Avec The Sleuth Kit
$ fls -r -p image.img # arborescence complète (deleted = *)
$ fls -r -p -d image.img # uniquement les fichiers supprimés
$ icat image.img 12345 > out.bin # récupérer par numéro d'inode/MFT
$ istat image.img 12345 # détails d'une entrée MFT

# Avec les outils Zimmerman (Windows)
PS> MFTECmd.exe -f C:\Image\%MFT --csv .\out --csvf mft.csv
PS> LogFileParser.exe -f C:\Image\LogFile -o logfile.csv
PS> UsnJrnl2Csv.exe -f C:\Image\%J -o usnjrnl.csv
```

A.6 Analyse ext4

```
# extundelete
$ sudo extundelete --restore-file 'chemin/relatif' /dev/sdb1
$ sudo extundelete --restore-all /dev/sdb1
$ sudo extundelete --restore-inode 12345 /dev/sdb1

# debugfs (e2fsprogs)
$ sudo debugfs /dev/sdb1
debugfs: lsdel # inodes récemment supprimés
debugfs: stat <12345> # détails d'un inode
debugfs: dump <12345> /tmp/out # extraire ses blocs
debugfs: cat <12345> # afficher (text)

# ext4magic
$ ext4magic /dev/sdb1 -M -d /recovery
```

A.7 TestDisk et PhotoRec

```
# Lancer TestDisk en mode interactif sur une image
$ sudo testdisk image.img

# Lancer PhotoRec (carving) sur une image
$ sudo photorec image.img

# PhotoRec en mode non-interactif (avancé)
$ sudo photorec /d /recovery/output /cmd image.img \
partition_none,fileopt,everything,enable,search
```

A.8 Vérification TRIM

```
# Windows
C:\> fsutil behavior query DisableDeleteNotify
DisableDeleteNotify (NTFS) = 0 <-- TRIM activé

# Linux
$ cat /sys/block/sdX/queue/discard_max_bytes
$ sudo fstrim -av # déclencher trim manuel
$ systemctl status fstrim.timer # voir si l'auto-trim tourne

# macOS
$ system_profiler SPSerialATADataType | grep -i 'TRIM Support'
```

A.9 RAID Linux (mdadm)

```
# Inspecter les disques
$ sudo mdadm --examine /dev/sd[a-d]1

# Assembler manuellement
$ sudo mdadm --assemble /dev/md0 /dev/sd[a-d]1

# Assembler en mode dégradé (utile en récupération)
$ sudo mdadm --assemble --force --run /dev/md0 /dev/sd[a-d]1

# Voir les détails
$ sudo mdadm --detail /dev/md0
$ cat /proc/mdstat
```

A.10 Memory et processus (forensique live)

```
# Capture de RAM Windows (DumpIt par MoonSols)
C:\> DumpIt.exe

# Capture RAM Linux
$ sudo dd if=/dev/mem of=memory.dump bs=1M
# (kernel récent : utiliser LiME ou AVML)

# Analyse avec Volatility 3
$ vol -f memory.dump windows.info
$ vol -f memory.dump windows.pslist
$ vol -f memory.dump windows.netscan
$ vol -f memory.dump windows.cmdline
```

ANNEXES**Chapitre B****Glossaire**

AES — Advanced Encryption Standard. Algorithme de chiffrement symétrique standardisé NIST en 2001. AES-128, AES-192 et AES-256 sont considérés comme résistant à toutes les attaques connues avec les moyens classiques actuels.

AFR — Annualized Failure Rate. Taux de panne annualisé d'un parc de disques. Calculé en projetant le nombre de pannes observées sur une fenêtre courte à une année complète.

APFS — Apple File System. Système de fichiers d'Apple introduit en 2017, copy-on-write, avec snapshots et support natif du chiffrement (FileVault).

Air-gap — Isolation physique d'un système ou support du réseau. Une bande de sauvegarde rangée dans un coffre est air-gappée ; un serveur connecté en VPN ne l'est pas.

BGA — Ball Grid Array. Boîtier de circuit intégré avec une matrice de billes de soudure sous le composant, courant pour les puces NAND et les contrôleurs SSD.

Btrfs — B-tree file system. Système de fichiers Linux copy-on-write avec snapshots et checksums, intégré au noyau depuis 2009.

Carving (data carving) — Reconstruction de fichiers à partir de signatures binaires dans les données brutes, sans utiliser les structures du système de fichiers. Voir chapitre 8.

Chain of custody — Chaîne de custody. Documentation continue de la prise en charge et de chaque manipulation d'une preuve numérique, essentielle en forensique judiciaire.

Chip-off — Désoudage physique d'une puce NAND pour la lire sur un programmeur dédié, indépendamment de son contrôleur d'origine.

CMR — Conventional Magnetic Recording. Mode d'écriture HDD avec pistes non-recouvrantes. Permet de réécrire n'importe quelle piste sans toucher aux voisines.

Copy-on-write — Stratégie d'écriture où toute modification écrit ailleurs et met à jour les pointeurs, sans écraser en place. Permet snapshots à coût marginal nul. Utilisé par APFS, Btrfs, ZFS.

DBIR — Data Breach Investigations Report. Rapport annuel publié par Verizon depuis 2008, référence statistique sur les compromissions de données.

ddrescue — GNU ddrescue. Outil d'imagerie bit-à-bit tolérant aux erreurs, avec gestion de mapfile pour reprises et passes ciblées.

DRAT / DZAT — Deterministic Read After Trim / Deterministic Zero After Trim. Garantit SSD que la lecture d'un LBA trimmé renvoie respectivement une valeur déterministe ou des zéros.

eMMC — embedded MultiMediaCard. Mémoire flash avec contrôleur intégré, format compact courant dans les smartphones et tablettes bas/moyen de gamme.

ECC — Error-Correcting Code. Codes correcteurs d'erreurs appliqués par les contrôleurs SSD à chaque page NAND pour corriger les bit-flips résiduels.

ext4 — Quatrième extended file system. Système de fichiers Linux par défaut depuis 2008, journalisé, avec support des extents pour la gestion efficace des gros fichiers.

FileVault — Chiffrement de volume macOS depuis Mac OS X 10.3 (FileVault 1) puis FileVault 2 depuis 10.7. Géré par la Secure Enclave sur les Mac Apple Silicon.

FTL — Flash Translation Layer. Couche logicielle dans le contrôleur SSD qui mappe les LBA logiques aux pages physiques NAND, gère le wear leveling et la garbage collection.

Garbage collection (GC) — Processus en arrière-plan du contrôleur SSD qui consolide les pages valides et applique la tension d'effacement physique sur les blocs marqués libres.

HAMR — Heat-Assisted Magnetic Recording. Technologie HDD qui chauffe ponctuellement la couche magnétique par laser pendant l'écriture pour réduire la taille des domaines stables. Commercialisée depuis 2024 sur les très gros disques.

Head swap — Transplantation de l'ensemble têtes/bras d'un HDD défaillant vers un boîtier identique en salle blanche. Intervention la plus courante en récupération physique HDD.

HDD — Hard Disk Drive. Disque dur magnétique mécanique.

Imagerie — Création d'une copie bit-à-bit d'un support vers un fichier image. Étape préalable à toute analyse ou récupération sérieuse.

ISO 14644-1 — Norme internationale qui définit les classes de salle blanche par concentration de particules par m³ d'air.

ISO 27037 — Norme internationale de référence pour l'identification, la collecte, l'acquisition et la préservation de preuves numériques.

JTAG — Joint Test Action Group. Protocole de débogage intégré dans la plupart des circuits modernes, exploité en récupération SSD pour accéder au contrôleur sans le désouder.

LBA — Logical Block Address. Adresse logique d'un secteur sur un support, exposée par l'interface SATA/NVMe. Le contrôleur SSD traduit les LBA en emplacements physiques via la FTL.

LUKS — Linux Unified Key Setup. Standard de chiffrement de volume sous Linux, implémenté par cryptsetup.

Mapfile — Fichier de carte utilisé par ddrescue pour enregistrer l'état exact de chaque secteur (copié, à recopier, lent, échoué). Permet la reprise et les passes ciblées.

MFT — Master File Table. Structure centrale de NTFS, un fichier qui contient une entrée de 1 024 octets par fichier et dossier du volume.

NAND flash — Mémoire à grille flottante, technologie sous tous les SSD, eMMC, UFS, microSD, clés USB.

NTFS — New Technology File System. Système de fichiers de Windows depuis NT, journalisé, avec attributs ACL, compression, chiffrement intégré et journaux \$LogFile / \$UsnJrnl.

Over-provisioning (OP) — Espace NAND réservé par le contrôleur SSD, invisible pour l'utilisateur, utilisé pour le wear leveling et la garbage collection. Typiquement 7 % consumer, 14-28 % entreprise.

PCB — Printed Circuit Board. Circuit imprimé extérieur d'un HDD ou SSD, contient le contrôleur et les composants d'interface.

PC-3000 — Plateforme matérielle et logicielle d'ACE Lab (Russie), standard de facto pour la récupération professionnelle HDD et SSD.

PMR — Perpendicular Magnetic Recording. Mode d'écriture HDD généralisé depuis 2005, domaines magnétiques orientés perpendiculairement à la surface.

RAID — Redundant Array of Independent Disks. Combinaison de plusieurs disques pour la performance, la résilience ou les deux. Niveaux courants : 0, 1, 5, 6, 10.

Ransomware — Logiciel malveillant qui chiffre les fichiers de la victime et exige une rançon pour la clé de déchiffrement.

Salle blanche / Cleanroom — Local à atmosphère contrôlée défini par ISO 14644-1. Pour la récupération HDD : ISO Class 5 (3 520 particules $\geq 0,5$ micron / m³).

SED — Self-Encrypting Drive. SSD qui chiffre automatiquement toutes les données via son contrôleur, généralement selon le standard TCG Opal.

SMR — Shingled Magnetic Recording. Mode d'écriture HDD avec pistes partiellement recouvrantes, gagne ~25 % de densité au prix d'une complexité firmware accrue.

Snapshot — Image instantanée d'un volume à un moment donné, à coût marginal nul sur les FS copy-on-write (APFS, Btrfs, ZFS).

TCG Opal — Standard du Trusted Computing Group pour le chiffrement matériel des SSD (Opal 1.0, Opal 2.0, Opalite, Pyrite).

TestDisk — Outil open source de référence pour la réparation de tables de partition et la récupération de partitions effacées.

TPM — Trusted Platform Module. Coprocesseur de sécurité soudé sur la carte mère, stocke des clés cryptographiques (BitLocker, mesure d'intégrité).

TRIM — Commande ATA (DATA SET MANAGEMENT) ou NVMe (DEALLOCATE) qui informe le contrôleur SSD des LBA libérés côté OS, permettant leur effacement physique.

UFS — Universal Flash Storage. Successeur d'eMMC pour les smartphones, plus rapide et avec architecture full-duplex.

Wear leveling — Stratégie du contrôleur SSD qui répartit les écritures sur toutes les cellules NAND pour qu'aucune ne s'use prématurément.

WORM — Write Once Read Many. Stockage immuable utilisé pour les sauvegardes protégées contre la suppression et la modification.

Write blocker — Dispositif matériel ou logiciel qui bloque toute écriture vers un support, garantissant l'intégrité forensique.

ZFS — Système de fichiers Sun Microsystems (2006), désormais OpenZFS. Copy-on-write, snapshots, checksums, self-healing sur volumes redondants.

ANNEXES**Chapitre C****Bibliographie**

Sources publiques effectivement consultées pour la rédaction de ce manuel, mai 2026. URLs simplifiées (préfixe <https://> et préfixe [www.](http://) omis).

Rapports d'industrie

Verizon Business	2025 Data Breach Investigations Report. Publié le 23 avril 2025.	verizon.com/about/news/2025-data-breach-investigations-report
Verizon Business	2025 DBIR Executive Summary (PDF).	verizon.com/business/resources/reports/2025-dbir-executive-summary.pdf
Backblaze	Drive Stats for 2025 (rapport annuel, 12 février 2026).	backblaze.com/blog/backblaze-drive-stats-for-2025/
Backblaze	Drive Stats Q1, Q2, Q3, Q4 2025.	backblaze.com/cloud-storage/resources/hard-drive-test-data
IBM Security	Cost of a Data Breach Report 2024.	ibm.com/reports/data-breach
ANOZR WAY	Fuites de données 2025 : +2,6 milliards de données compromises. Janvier 2026.	anozrway.com/fr/blog/fuites-de-donnees-en-2025/

Normes et standards

ISO	ISO 14644-1:2015 — Cleanrooms and associated controlled environments — Part 1: Classification of air cleanliness by particle concentration.	iso.org/standard/53394.html
ISO	ISO/IEC 27037:2012 — Guidelines for identification, collection, acquisition and preservation of digital evidence.	iso.org/standard/44381.html
Trusted Computing Group	TCG Storage Architecture Core Specification, Opal 2.0.	trustedcomputinggroup.org/resource/storage-work-group-storage-security-subsystem-class-opal/

Salle blanche et récupération HDD

DriveSavers	Certified ISO Class 5 Cleanroom.	drivesaversdatarecovery.com/why-us/certified-iso-class-5-cleanroom/
SalvageData	ISO 5 Cleanroom — Certified Data Recovery.	salvagedata.com/about/certified-data-recovery/cleanroom
Gillware	Clean Room Data Recovery.	gillware.com/hard-drive-data-recovery/data-recovery-clean-room/
Secure Data Recovery	Class 10 ISO 4 Cleanroom.	securedatarecovery.com/certifications/cleanroom
Rossmann Group	CMR vs SMR: How Recording Technology Affects Recovery. 2026.	rossmanngroup.com/technical-reference/cmr-vs-smr-hard-drives
HackMag	Unmasking Shingled Magnetic Recording in Western Digital and Seagate HDDs. 2025.	hackmag.com/security/hdd-smr

Wikipedia	Shingled magnetic recording.	en.wikipedia.org/wiki/Shingled_magnetic_recording
------------------	------------------------------	---

SSD, NAND, TRIM

Rossmann Group	What TRIM Does and Why It Destroys Data.	rossmanngroup.com/technical-reference/what-trim-does-and-why-it-destroys-data
Rossmann Group	TRIM & Garbage Collection: When SSD Data Is Gone.	rossmanngroup.com/services/ssd-data-recovery/trim-garbage-collection
DataCare Labs	SSD TRIM, Garbage Collection, and Write Amplification. 2025.	datacarelabs.com/blog/ssd-trim-garbage-collection-deleted-files/
Seagate	What Are SSD TRIM and Garbage Collection?	seagate.com/blog/what-are-ssd-trim-and-garbage-collection/
Kingston	The Importance of Garbage Collection and TRIM.	kingston.com/en/blog/pc-performance/ssd-garbage-collection-trim-explained
Lexar Enterprise	Comparing NAND Flash Technology: SLC, MLC, TLC, and QLC. 2026.	lexarenterprise.com/comparing-nand-flash-slc-mlc-tlc-qlc-industrial-application/
TechTarget	Explore benefits, tradeoffs with SLC vs MLC vs TLC.	techtarget.com/searchstorage/tip/The-truth-about-SLC-vs-MLC
Belkasoft Forensic Focus	Recovering Evidence from SSD Drives: Understanding TRIM, Garbage Collection and Exclusions.	forensicfocus.com/articles/recovering-evidence-from-ssd-drives-in-2014/
DiskGenius	SSD Data Recovery Explained. 2025.	diskgenius.com/resource/ssd-data-recovery-explained.html

Systèmes de fichiers

Sygnia	The Forensic Value of MFT Slack Space in Modern Windows Systems. 2025.	sygnia.co/blog/the-forensic-value-of-mft-slack-space/
Mahmoud Shaker / DFIR-Notes	Master File Table (MFT), NTFS, \$LogFile, and \$UsnJrnl: Forensics. 2025.	mahmoud-shaker.gitbook.io/dfir-notes/
ScienceDirect	Master File Table - an overview (chapter from File System Forensic Analysis, Brian Carrier).	sciencedirect.com/topics/computer-science/master-file-table
Number Analytics	Unlocking Ext4: A Forensic Guide. 2025.	numberanalytics.com/blog/ultimate-guide-ext4-digital-forensics
Botmonster Tech	Linux File Recovery: extundelete, PhotoRec, Btrfs snapshots. 2026.	botmonster.com/posts/linux-file-recovery-undelete-ext4-btrfs/
extundelete project	extundelete: An ext3 and ext4 file undeletion utility.	extundelete.sourceforge.net/

Outils

GNU project	GNU ddrescue manual et code source.	gnu.org/software/ddrescue/
CGSecurity	TestDisk & PhotoRec.	cgsecurity.org/wiki/TestDisk
Sleuth Kit	The Sleuth Kit and Autopsy.	sleuthkit.org/
Eric Zimmerman	Tools (MFTECmd, RECmd, etc.).	ericzimmerman.github.io/

Maxim Suhanov dfir_ntfs. github.com/msuhanov/dfir_ntfs

Sauvegarde et prévention

Veeam	3-2-1 Backup Rule Explained. 2025.	veeam.com/blog/321-backup-rule.html
Veeam Community	The 3-2-1-1-0 Rule in Practice. 2026.	community.veeam.com/blogs-and-podcasts-57/the-3-2-1-1-0-rule-in-practice
Object First	3-2-1-1-0 Backup Rule: How Object First and Veeam Implement It. 2025.	objectfirst.com/blog/how-object-first-and-veeam-bring-3-2-1-1-0-to-life/
Opti9 Tech	The 3-2-1-1-0 Backup Strategy Explained. 2025.	opti9tech.com/blog/the-3-2-1-1-0-backup-strategy-explained/

Études de cas

Control Engineering	Throwback Attack: How NotPetya Ransomware Took Down Maersk. Mise à jour août 2025.	controleng.com/throwback-attack-how-notpetya-accidentally-took-down-global-shipping-giant-maersk/
CSO Online	Rebuilding after NotPetya: How Maersk moved forward. 2019.	csoonline.com/article/567845/rebuilding-after-notpetya-how-maersk-moved-forward.html
Redmondmag	Inside a Domain Controller Nightmare (Maersk).	redmondmag.com/blogs/scott-bekker/2018/08/domain-controller-nightmare.aspx
Red Goat Cyber Security	Maersk incident response.	red-goat.com/why-you-should-test-your-incident-response-a-review-of-the-maersk-incident/
Threatpost	Hacker Puts Hosting Service Code Spaces Out of Business. Juin 2014.	threatpost.com/hacker-puts-hosting-service-code-spaces-out-of-business/106761/
eSecurity Planet	Code Spaces Destroyed by Cyber Attack. 2014.	esecurityplanet.com/networks/code-spaces-destroyed-by-cyber-attack/
InfoWorld	Murder in the Amazon cloud (Code Spaces). 2014.	infoworld.com/article/2179073/murder-in-the-amazon-cloud.html
breaches.cloud / Wiz	Codespaces (2014) - Public Cloud Security Breaches.	breaches.cloud/incidents/codespaces/
The Hacker News	Cyber Attack On Code Spaces Puts Hosting Service Out of Business. 2014.	thehackernews.com/2014/06/cyber-attack-on-code-spaces-puts.html

Sources complémentaires DBIR

Halcyon	Verizon DBIR Shows Ransomware Involved in 44% of Data Breaches. 2025.	halcyon.ai/blog/verizon-dbir-shows-ransomware-involved-in-44-of-data-breaches
SpyCloud	Breaking Down the 2025 Verizon Data Breach Investigations Report.	spycloud.com/blog/verizon-2025-data-breach-report-insights/
Keepnet Labs	2025 Verizon DBIR: Key Facts, Trends & Statistics. Mise à jour mars 2026.	keepnetlabs.com/blog/2025-verizon-data-breach-investigations-report
Rhymetec	The Verizon Data Breach Report 2025: Key Takeaways & Statistics.	rhymetec.com/the-verizon-data-breach-report-2025-key-takeaways-statistics/

Sources que les corpus précédents ont citées sans qu'elles soient vérifiables

Pour transparence, les références suivantes sont régulièrement citées dans la littérature grand public sur la récupération de données, mais je n'ai pas pu les vérifier indépendamment et elles n'ont pas servi de source à ce manuel :

- Études comparatives chiffrées de logiciels de récupération publiées sur les sites des éditeurs ou de leurs affiliés.
- « Rapports internes de laboratoires » Ontrack, Kroll, SalvageData, Dafotec avec des taux de succès numériques par technique.
- Numéros précis de versions de PC-3000 (les versions existent, mais ACE Lab ne publie pas de roadmap publique détaillée par version).
- Études académiques mentionnées sans DOI ni référence bibliographique vérifiable.

Fin du manuel. Mai 2026.