

Qu'est-ce qu'un système de fichiers ?

Un sacré bazar

40 • 30 

COMPOSANTS

🕒 11 MIN



Par Vincent Hermann
Le vendredi 23 juin 2023 à 10:47



 Signaler une erreur  Offrir

Le système de fichiers est l'un de ces multiples éléments invisibles qui permettent à un appareil disposant d'un stockage de fonctionner. Mais en quoi consistent-ils ? Quelles sont leurs fonctions ? Ont-ils évolué ? Voici l'essentiel de ce qu'il faut savoir sur le sujet.

Un système de fichiers s'intercale entre le système d'exploitation et le matériel. On le définit habituellement comme un système de classement des données, ce qui suppose que tout a été pensé : la structure des informations, leur organisation, la manière dont l'écriture et la modification vont se faire, ce qui se passe en cas de suppression, etc.

Surtout, un système de fichiers – que l'on retrouve souvent sous le sigle anglais FS, pour *file system* – va permettre à une personne d'accéder à ses données. Et non seulement elle doit pouvoir y accéder, mais cette livraison d'informations doit se faire le plus vite possible.

La question des performances est certes importante, mais pas nécessairement prioritaire. Comme on le verra, il existe des systèmes spécialisés ayant des spécificités, par exemple sur l'intégrité ou la



Mais avant d'aller plus loin, revenons aux bases.

Et la mémoire de masse fut

Si l'on pouvait, toutes les données seraient dans la mémoire vive. Les opérations y sont bien plus rapides. Pour des questions évidentes de coût, ce n'est pas possible. Il faut donc un autre type de stockage, une mémoire persistante, secondaire, dans lequel on va pouvoir entreposer la masse toujours croissante de données pour y accéder en cas de besoin.

Pour résumer à grands traits, ce fut le rôle des disquettes, puis des disques durs et plus récemment des SSD (ainsi que des CD, DVD et autres). La technologie utilisée n'a pas d'importance ici, on retiendra surtout ceci : pour gérer ces grands espaces, il fallait une organisation capable de répondre à tous les scénarios. En plus des éléments cités plus haut, un « FS » s'occupe ainsi de la convention de nommage, c'est-à-dire la manière dont sont nommés les fichiers. Les permissions, règles d'accès, quotas, privilèges ou encore les métadonnées sont aussi de son ressort.

- **L'extraordinaire évolution du stockage**

C'est donc une composante primordiale d'un appareil et l'une des plus discrètes. L'analogie la plus simple est celle d'une grande salle d'archivage. Et qui dit archivage dit index, pour retrouver les documents dès que l'on en a besoin.

Dans la grande majorité des cas, on parle donc de fichiers et de dossiers comme organisation élémentaire, ce qui apparaît devant nos yeux comme des arborescences de dossiers dans le gestionnaire de fichiers du système d'exploitation. Il s'agit d'une vue abstraite et on ne la voit pas toujours, car tous les appareils ne laissent pas forcément un accès direct à cette structure. Un exemple ? iOS ne le permettait pas avant sa version 11 et l'apparition de Fichiers.

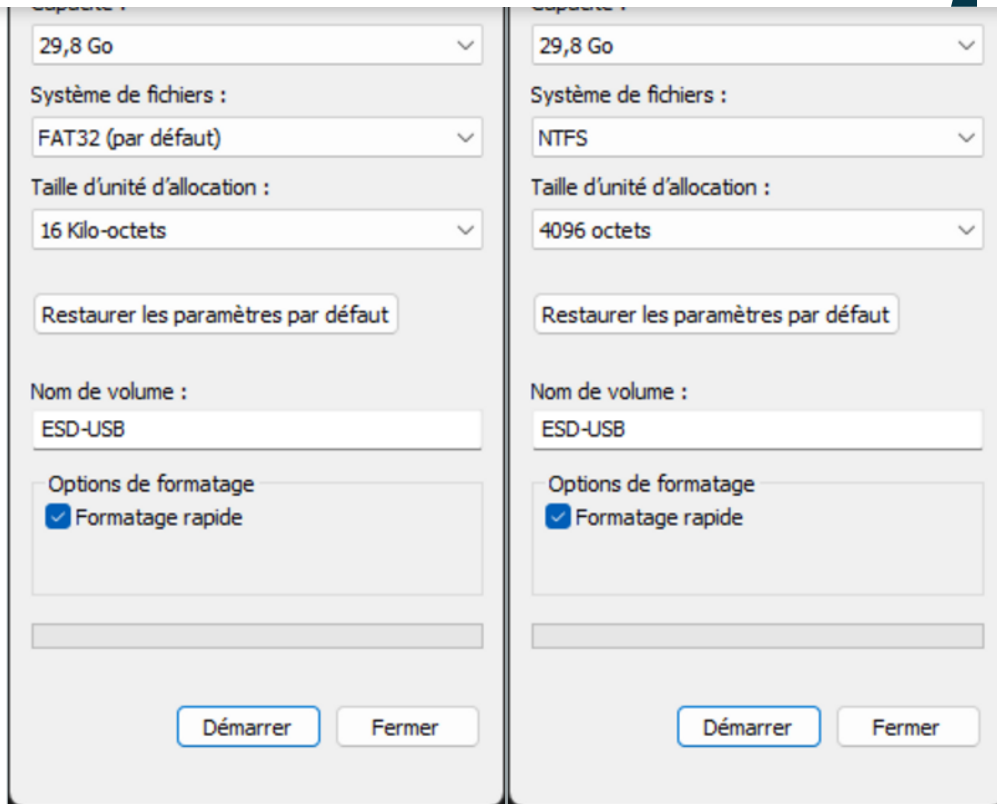
Sachez d'ailleurs que les dossiers n'ont pas existé tout de suite. Chez Microsoft par exemple, ils sont arrivés avec MS-DOS 2.0.

Le lien avec le matériel

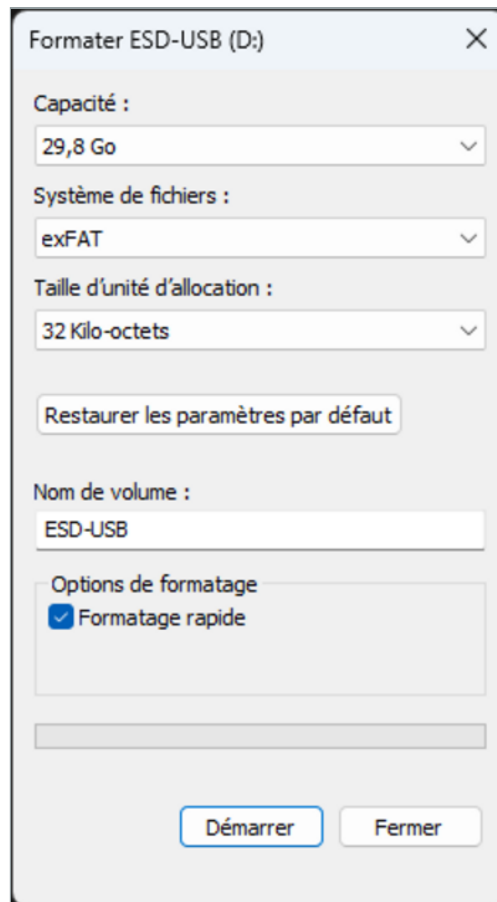
De nombreux utilisateurs connaissent bien l'expression « formater » un disque. Pour beaucoup, c'est le synonyme de « tout effacer et recommencer », avec l'installation d'un système d'exploitation neuf. Ce n'est toutefois pas le sens premier du verbe, mais sa conséquence.

« Formater » signifie littéralement « mettre au format ». C'est l'action qui consiste à établir sur le disque de stockage la structure qui va ensuite accueillir les informations. Parmi les caractéristiques de ce format, on retrouve notamment la taille du bloc, c'est-à-dire la taille de la plus petite unité que le FS va gérer.

À la manière d'une HAL (hardware abstraction layer), le système de fichiers est une abstraction du matériel. Les utilisateurs interviennent très peu souvent sur la taille des blocs (appelés *clusters* en anglais). Le cas le plus fréquent est sans doute le formatage d'une clé USB. Sous Windows par exemple, le choix par défaut est la FAT32, qui propose des blocs de 16 ko. Mais on peut également la formater en NTFS ou exFAT, qui donnent respectivement des blocs de 4 ou 32 ko par défaut. Dans tous les cas, le panneau Formater permet de choisir une autre taille.



Two side-by-side screenshots of the Windows 'Format' dialog box for an ESD-USB drive. The left screenshot shows the 'Formatage rapide' option checked and the file system set to FAT32. The right screenshot shows the file system set to NTFS.



Format ESD-USB (D:) X

Capacité :
29,8 Go

Système de fichiers :
exFAT

Taille d'unité d'allocation :
32 Kilo-octets

Restaurer les paramètres par défaut

Nom de volume :
ESD-USB

Options de formatage
 Formatage rapide

Démarrer Fermer

La taille de ces blocs a une importance capitale et celle par défaut l'est parce qu'elle représente le meilleur compromis entre performances et consommation de l'espace. Elle peut changer également en fonction de l'espace adressé. **Avec NTFS** par exemple, la taille par défaut est de 4 ko, mais elle est de 2 ko pour les espaces de moins de 2 Go et de 8 ko pour 16 To ou plus.



utilise si le fichier grossit. Si vous avez un fichier de 11 ko, il utilisera trois blocs : deux complets et un troisième rempli aux trois quarts (3 ko utilisés sur 4 disponibles). Un fichier occupera toujours un nombre entier de blocs.

Selon le système de fichiers et avec le temps, les données peuvent être écrites « là où il y a de la place ». Les blocs d'un fichier peuvent donc être situés aux quatre coins d'un disque, phénomène que l'on nomme fragmentation et qui a fait les beaux jours de certains outils à une époque. Pour un disque dur, c'était un vrai problème : la tête de lecture devait se balader un peu partout pour recoller les morceaux, faisant baisser les performances générales. Un problème en grande partie résolu avec les SSD.

De manière générale, la taille des blocs grandit avec l'espace à adresser, que l'on appelle le plus souvent partition ou volume en fonction des cas (mais ce n'est pas toujours si simple). Il y a toutefois des exceptions. L'exFAT a ainsi des blocs de 32 ko parce qu'elle a été spécifiquement conçue pour les cartes SD des appareils photo numériques. Les photos sont souvent des tailles importantes et des petits blocs ne sont pas nécessaires.

On en revient donc au compromis. Une grande taille de bloc permettra une écriture plus rapide des données puisque le nombre de blocs à gérer pour un gros fichier sera plus petit, avec une fragmentation nécessairement plus faible. En revanche, si l'on utilise souvent des petits fichiers, l'espace perdu sera plus important. Une problématique que nous avons déjà abordée avec la taille des blocs « physiques » des disques durs (512 o ou 4 ko).

Il faut donc bien distinguer deux choses. D'un côté l'écriture « logique » quand le système d'exploitation place les données sur la partition, avec son système de fichier. De l'autre, l'écriture physique quand le périphérique de stockage place les données sur ses plateaux.

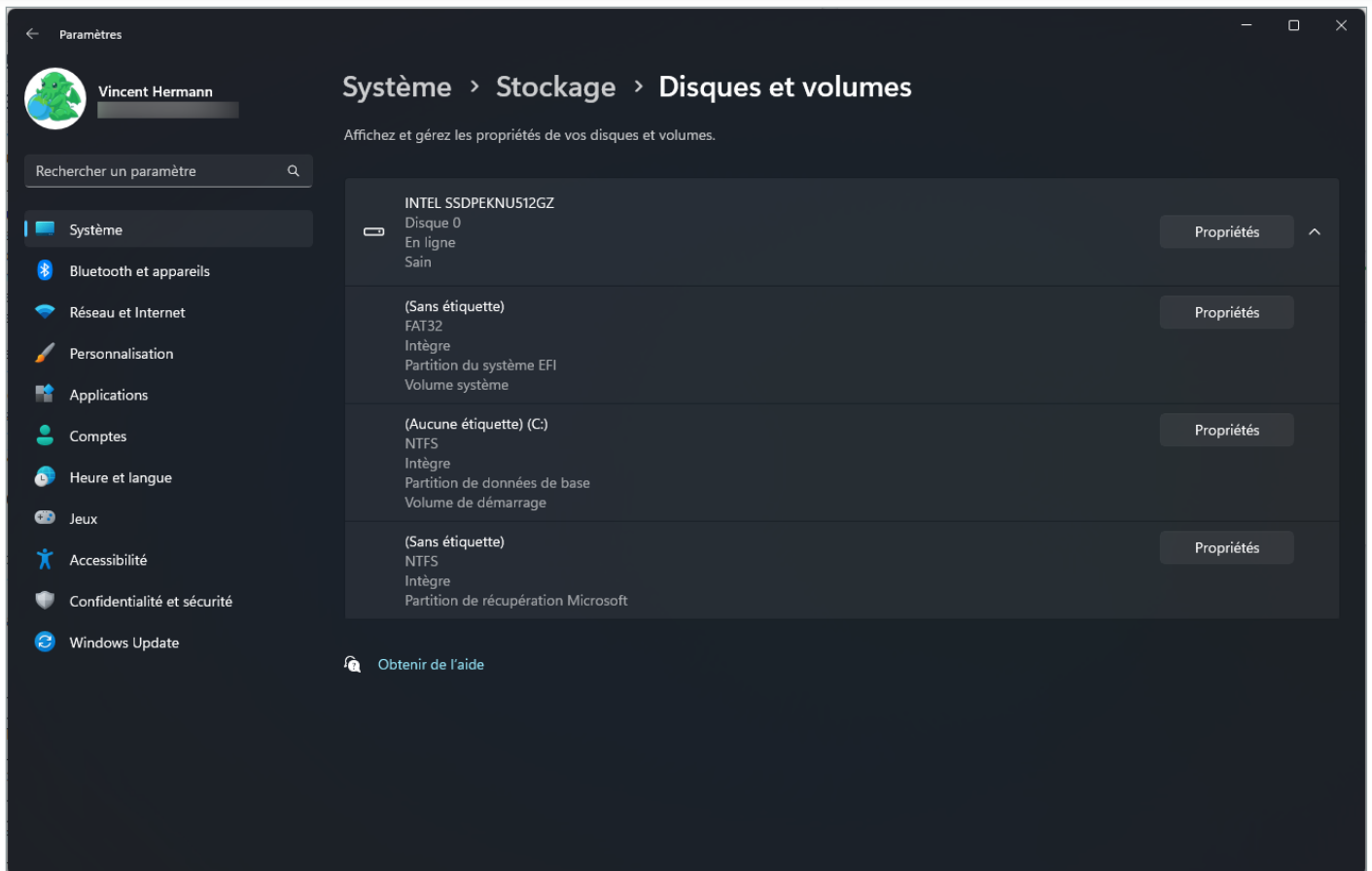
- **Composition, technologies et fonctionnalités : on vous dit tout du disque dur moderne**

Partition et volume, sans musique

Comme beaucoup le savent, il est possible de découper un disque en plusieurs parties. Dans la grande majorité des cas, on parle de partitions : une division logique du disque physique. Il est très courant par exemple de diviser un gros disque dur en plusieurs partitions (primaire, logique et étendue), chacune dédiée à un usage particulier. Chaque partition reçoit généralement une lettre (à partir de C) pour l'identifier et peut avoir un système de fichiers différent. Certaines peuvent être masquées, comme celles d'amorçage et de restauration, selon les systèmes d'exploitation.

Le volume, quant à lui, est une appellation plus souple désignant une unité logique de stockage de données. N'est-ce pas exactement la définition d'une partition ? Oui... et non. Le volume prend en fait place sur un disque dynamique, c'est-à-dire une structure logique pouvant intégrer plusieurs disques physiques. Leur espace est agrégé en une réserve commune, sur laquelle les volumes sont définis. Par exemple, si l'on a trois disques de 2 To, on peut créer un volume de 6 To, deux de 3 To, un de 4 To, etc. De nombreux systèmes modernes savent créer ce type de structure.

Il s'agit bien d'une virtualisation du stockage, avec tous les avantages de flexibilité que l'on peut en attendre. Par exemple, les disques peuvent être configurés en RAID, pour des besoins de sécurité et/ou de performances. Le mécanisme est également omniprésent dans les serveurs, puisque



Volumes de disque sous Windows 11

Un mot enfin sur deux sigles que vous avez peut-être croisés au moment de partitionner : **MBR** et **GPT**.

Le premier signifie Master Boot Record et sert globalement d'index au début du disque dur pour renseigner notamment sur son découpage et l'amorce du système d'exploitation (boot loader). On peut y penser comme une forme de table des matières.

Seulement voilà, il s'agit d'une très ancienne norme, car elle a été créée par IBM en 1983. Le MBR a plusieurs limitations, dont une compatibilité limitée avec le matériel moderne, ne pouvant par exemple pas gérer de disque de plus de 2 To. Il ne peut gérer également que quatre partitions primaires. Si l'on en souhaite plus, il faut garder un des emplacements pour une partition étendue, dans laquelle on pourra créer autant de partitions logiques que souhaité.

La GPT signifie quant à elle GUID Partition Table. On peut créer théoriquement autant de partitions que l'on veut, chacune étant munie d'un identifiant unique appelé GUID, pour Globally Unique Identifier. En pratique, le nombre et la taille des partitions dépend des limitations de chaque système d'exploitation. Les cartes mères UEFI prennent en charge le partitionnement GPT.

Le grand bazar

Avec une telle quantité d'informations, comment s'y retrouver ? Cette partie est un peu plus technique, mais nous allons tâcher de la vulgariser autant que possible.



table) que l'on retrouve notamment dans les FAT16 et FAT32. Dans ce format, chaque fichier stocké dispose d'un numéro unique. La table agit comme un index et fait le lien entre ce numéro et l'identifiant du premier bloc du fichier. Cet index étant unique, il faut lui réserver un espace dédié sur le disque, expliquant pourquoi l'espace disponible d'un disque ainsi formaté est toujours inférieur.

Bon, mais puisque l'on a seulement l'identifiant du premier bloc, comment faire pour trouver le reste ? Il y a plusieurs manières de procéder, mais on retiendra la plus courante : chaque bloc contient à la fin l'identifiant du prochain bloc. S'il s'agit du dernier, il contient la valeur EOF, pour *End of file*, signalant qu'il n'y a plus d'autre bloc. La méthode a l'avantage d'être très simple dans son approche, mais très sensible à la fragmentation. Les anciens systèmes de fichiers comme FAT16 et FAT32 ne disposent pas de mécanismes dédiés à cette fragmentation. Ils sont arrivés plus tard dans les versions successives de NTFS. Sur les systèmes UNIX/Linux, les systèmes de fichiers ont depuis longtemps de tels mécanismes.

L'autre grande méthode pour retrouver ses données est le stockage indexé. Plus de table d'allocation ici, chaque fichier est constitué d'un bloc d'index servant à stocker les informations sur ceux de données. Avantage immédiat : plus besoin de déterminer à l'avance un grand index pour référencer les fichiers, l'espace formaté étant très proche de l'espace matériel total. Désavantage tout aussi immédiat : l'espace sera plus vite consommé, puisque chaque fichier a toujours besoin d'un bloc supplémentaire. Pour reprendre l'exemple du fichier de 1 ko sur des blocs de 4 ko, il faudrait alors deux blocs, donc 8 ko.

Bon, et c'est tout ?

Pas tout à fait, car sur les systèmes de fichiers modernes, les fonctions vont bien au-delà de la simple organisation des données. Nous en avons parlé par exemple, mais la convention de nommage fait partie des attributions des FS. La première FAT de Microsoft permettait par exemple des noms de fichiers composés de huit caractères, suivis d'un point puis d'une extension de trois caractères. Cette limitation a sauté avec la FAT32 (via **LFN**) et n'est plus d'actualité depuis bien longtemps dans aucun système courant.

Parmi les autres fonctions qu'un système de fichiers peut prendre en charge, on trouve :

- Le chiffrement, intégral ou non
- Les clichés instantanés (snapshots) pour les fichiers ou le disque entier, c'est-à-dire la sauvegarde à un instant T d'un lot de données en vue d'une éventuelle restauration
- La journalisation de tous les changements intervenant sur le disque, pour restaurer des données en cas de panne ou d'erreur
- La gestion des autorisations et permissions pour l'accès aux données
- Le clonage de fichiers, copie quasi instantanée sur laquelle seules les données modifiées par rapport au fichier d'origine sont sauvegardées
- La copie sur écriture, mécanisme lié au précédent, qui permet la création automatique d'une nouvelle copie du fichier dès que des modifications sont opérées
- Le partage d'espace, permettant à plusieurs volumes de croître et décroître dynamiquement selon les besoins, sans toucher aux partitions
- Le contrôle d'intégrité des données (via des checksums)



AFPS ou encore ZFS. Comme on verra, les approches peuvent changer radicalement de l'un à l'autre.



 Signaler une erreur

 Offrir

40 commentaires



Bloptimus - 23/06/23 à 11:02:02

#1

Joli ! Merci ;)



Wosgien - 23/06/23 à 11:13:07

#2

Bonne introduction.

Il faudra aborder la journalisation, c'est très mal compris même par des informaticiens, et pourtant c'est ce qui permet d'avoir une confiance presque absolue dans le FS.

Et bêtement, parler des FS comme ceux des VAX, Palm (oui, les palm de mémoire sont proche d'un mainframe), mainframe IBM est intéressant: ils ont des notions de taille et extension du fichier qui ne sont plus du domaine de la BDD mais manquent parfois à nos FS.

Utiliser une VM stockée sur un FS "classique" a ses limites.

Je rattacherais dans la liste des fonctionnalités les snapshots du clônage + copie sur écriture car ils fonctionnent sur ce principe! C'est pour cela qu'ils sont "instantanés".

Et je m'amuse toujours à montrer que certaines "nouveauités" dans les FS sont en fait là depuis longtemps et disponible à tout le monde depuis 2000/XP (par exemple le snapshot, très utilisé par moi-même pour annuler des modifs depuis près de 20ans sous Windows) - ou le versioning de fichier quasi obligatoire sur ODS sur VAX.

Et mon petit chouchou: si, il fallait aussi défragmenter les disques sous Linux - et ça ne pouvait être fait que si le disque était en lecture seule contrairement à Windows qui le fait à chaud depuis longtemps (mais bon, c'est de l'histoire ancienne avec les SSD).

Il y a aussi le futur, avec les SSD en zoning histoire de ne plus se farcir l'historique des disques durs dans les SSD - et l'abstraction constante qui fait que même sur un disque dur moderne, l'écriture n'a pas lieu là où on le croit...



Wosgien - 23/06/23 à 11:17:06

#3

Un petit détail sur les FAT en 16/32Ko.

A une époque lointaine, on avait une perte énorme de place si on passait en 32ko, beaucoup de fichiers étant petits.

Les systèmes de compression de disque étaient d'autant plus efficaces qu'il récupéraient cette place perdue car ils savaient mettre les fichiers "bout à bout".

C'est toujours le cas avec la compression de répertoire sous Windows - utile pour les petits fichiers en lecture seule et données froides.



Super article :) !

 **xlp** - 23/06/23 à 11:46:05 #5

↩ **Wosgien** Très curieux des snapshots dispos depuis Windows 2000/XP. Tu parles de VSS?

 **alex.d.** - 23/06/23 à 11:54:58 #6

Je n'avais jamais réalisé à quel point les tailles de blocs par défaut sous Windows sont énormes. Sous Linux, c'est 4 ko par défaut.

 **Patch** - 23/06/23 à 11:58:15 #7

▮ Chaque partition reçoit généralement une lettre (à partir de C)

Uniquement pour Windows. Sur Linux (et probablement tous les Unix/Unix like), elles démarrent toujours à a.


Et l'explication vient du fait que A et B étaient destinés aux lecteurs de disquettes.

 **Goldoark** - 23/06/23 à 12:21:42 #8

Merci beaucoup. N'oubliez pas ReFS...

 **Duleunien** - 23/06/23 à 12:25:43 #9

Est-ce que tu vas parler de GPFS ?

 **Vekin** - 23/06/23 à 13:21:48 #10

↩ **Wosgien** Pour les snapshots (Shadow Copy), je suis d'accord, les points de restauration sauvent la mise !

Malheureusement, j'ai eu parfois la surprise de constater que l'espace alloué pour les points de restauration était défini à zéro pourcent... quand j'en avais le plus besoin, bien entendu... alors maintenant je prends l'habitude d'aller vérifier de temps en temps.

Concernant les versions précédentes, j'étais convaincu pendant longtemps que ça devait se faire tout seul, activé par défaut, mais en fait, non, il faut le faire manuellement.

Édité par Vekin le 23/06/2023 à 13:22



Votre commentaire

Connecté en tant que **TheBigBug**



Commentaire...



Envoyer 

2000 - 2023 INpact MediaGroup - SARL de presse, membre du SPIIL. N° de CPPAP 0326 Z 92244.

Marque déposée. Tous droits réservés. [Mentions légales et contact](#)