CLOUDFLARE

Rapport 2024 consacré à la gestion et à la sécurité des API

Sommaire

Cliquez sur la section pour passer directement à la page désirée.

03 **Synthèse**

04 **Instantané : le trafic lié aux API à travers le monde**

05 **Conclusions principales**

06 **Surfaces d'attaque dissimulées**

07 Le risque lié aux API fantômes

08 **Erreurs courantes relatives aux API**

09 Le risque lié au mauvais diagnostic des erreurs concernant les API

10 **Principales vulnérabilités en matière de sécurité des API**

11 Le rôle des vulnérabilités des API lors d'une attaque MDM

12 Deux moyens d'atténuer les vulnérabilités courantes affectant les API

13 **Le monde orienté API**

13 Tendances régionales

14 Pics de trafic au Moyen-Orient

15 Le trafic lié aux API ralentit-il l'ensemble ?

16 Le trafic lié aux API en fonction des secteurs

17 Indicateurs sectoriels

18 **Prévisions pour 2024 et au-delà**

23 **Recommandations**

30 **Annexes**

30 Glossaire de la sécurité des API

32 Descriptions des codes de statut HTTP

33 Notes de fin

Synthèse

Le réseau Internet est un flux incessant de conversations entre ordinateurs. Ces conversations sont souvent conduites à l'aide d'interfaces de programmation (Application Programming Interface, API), qui nous permettent d'interagir de manière nouvelle avec les logiciels et les applications. Pour prendre un exemple, l'API ChatGPT d'OpenAI permet à Slack de rationaliser les flux de travail basés sur la discussion en direct et à Booking.com de proposer une expérience de planification de voyage plus personnalisée.

De nos jours, le trafic lié aux API surclasse les autres formes de trafic Internet, **en totalisant plus de la moitié (57 %) du trafic dynamique** traité par Cloudflare[1] l'année dernière.

Toutefois, comme détaillé dans le présent **rapport 2024 consacré à la gestion et à la sécurité des API**, ces dernières sont de plus en plus complexes à gérer et à protéger contre les abus.

De nombreuses entreprises manquent d'informations précises sur leurs API, par exemple. Cloudflare a ainsi découvert 30,7 % de points de terminaison d'API supplémentaires via l'identification par apprentissage automatique (Machine Learning) par rapport à ce que les entreprises ont elles-mêmes rapporté.[2]

30,7 %

de points de terminaison d'API supplémentaires

Malheureusement, les entreprises ne peuvent se défendre correctement contre ce qu'elles ne peuvent voir.

Celles qui mettent en place des mesures de sécurité des API sans cartographie précise et en temps réel de leur paysage en matière d'API **peuvent bloquer involontairement le trafic légitime.**

Prenons l'exemple du **code d'erreur « Too Many Requests » (429, Trop de requêtes)**, **soit la catégorie d'erreurs du client d'API la plus atténuée** par Cloudflare en 2023. L'émission d'un code 429 n'implique pas automatiquement *qu'un trop grand nombre de requêtes ont été envoyées par un acteur malveillant*. Si les limites de volume de requêtes responsables des erreurs ont été initialement mises en place à la suite d'une attaque par déni de service distribué (Distributed Denial of Service, DDoS), le fait d'imposer des limites de volume trop larges et inappropriées peut toujours bloquer les utilisateurs légitimes. (Il convient ici de noter que **la protection contre les attaques DDoS constituait la première méthode d'atténuation des API** pour les clients Cloudflare.)

Ce rapport a pour objectif de proposer un précieux cadre de référence pour les entreprises qui cherchent à **évaluer l'intégrité de leur gestion des points de terminaison d'API de manière globale**. Après tout, la sécurité des API doit également incorporer des données permettant de gérer la visibilité, les performances et les risques.

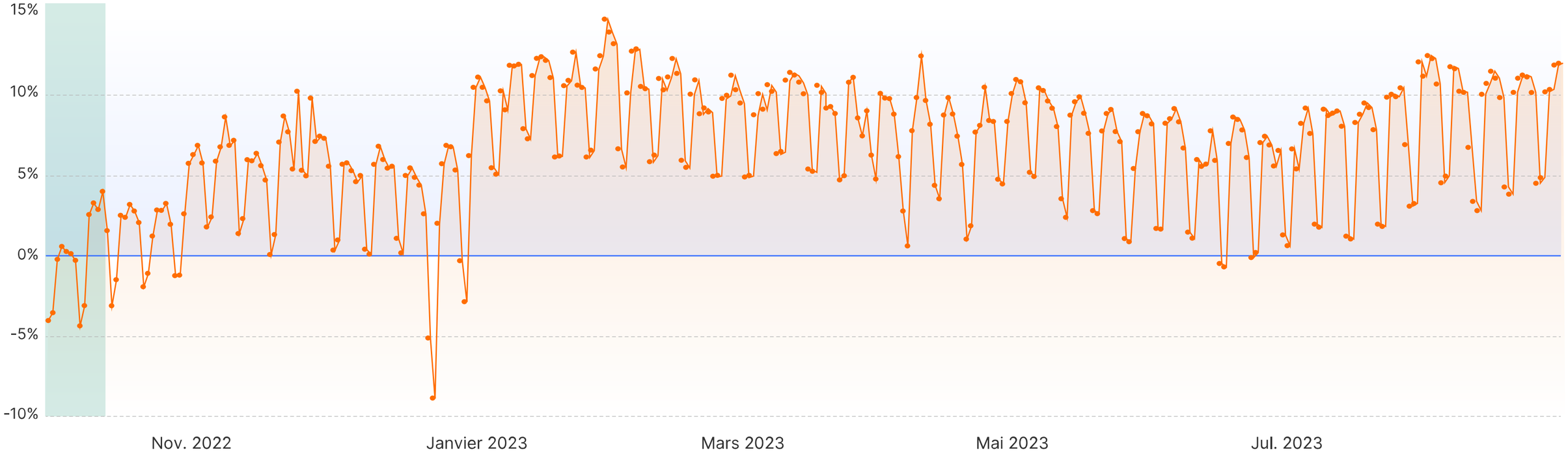
MÉTHODOLOGIE

Les conclusions de ce rapport se basent sur les schémas de trafic agrégé observés par le réseau mondial de Cloudflare (notamment les services de pare-feu d'applications web, de protection contre les attaques DDoS, de gestion des bots et de passerelle d'API proposés par Cloudflare) entre le 1er octobre 2022 et le 31 août 2023. Cloudflare traite en moyenne plus de 50 millions de requêtes HTTP par seconde et bloque quotidiennement 170 milliards de cybermenaces (là encore en moyenne).

Instantané : le trafic lié aux API à travers le monde

Croissance mondiale du trafic des API au fil du temps

La base de référence est mise en valeur, codes de réponse 200 et réponse de cache dynamique uniquement.



Entre le 1er octobre 2022 et le 31 août 2023, le trafic d'API assorti de réponses fructueuses (code de statut 200) représentait entre 53,1 % et 60,1 % du trafic HTTP dynamique de Cloudflare. Le contenu dynamique désigne le contenu qui change en fonction de facteurs spécifiques à l'utilisateur, comme l'heure de la visite, son emplacement géographique et son appareil.

Conclusions principales

Le trafic lié aux API surclasse les autres formes de trafic Internet

Les requêtes d'API fructueuses totalisaient **57 % du trafic Internet** (trafic HTTP dynamique) traité par Cloudflare.[1]

Surfaces d'attaque inconnues

Les modèles de Machine Learning ont identifié près d'**un tiers (30,7 %) de points de terminaison d'API supplémentaires** que ce que les entreprises ont elles-mêmes rapporté.[2]

Première erreur : Trop de requêtes

Plus de la moitié **(51,6 %) des taux d'erreur d'API se composaient d'erreurs « Too Many Requests »** (Trop de requêtes, c'est-à-dire d'erreurs 429).[3]

Première méthode d'atténuation : la protection contre les attaques DDoS

Un tiers **(33 %) des mesures d'atténuation des menaces liées aux API comprenaient le blocage d'attaques DDoS**.[4]

Variations sectorielles

Les secteurs présentant la **part la plus élevée de trafic lié aux API étaient ceux de l'IdO, des trains, bus et taxis, des services juridiques, des jeux/du multimédia, ainsi que ceux de la logistique/de l'approvisionnement**.[5]

Variations régionales

La part de trafic lié aux API a été la plus élevée **en Afrique et en Asie. C'est au Moyen-Orient que le trafic des API a le plus varié**.[6]

Surfaces d'attaque dissimulées

Pour les entreprises, les API sont source d'avantages concurrentiels : veille économique plus efficace, déploiements cloud plus rapides, intégration de nouvelles capacités IA, etc. Toutefois, la première étape vers l'optimisation des API consiste à disposer d'un inventaire complet des noms d'hôte et de l'ensemble des points de terminaison d'API exposés à Internet.

Une entreprise ne peut pas gérer ni protéger une API si elle ne sait pas que cette dernière existe. **Il s'avère d'ailleurs que de nombreuses entreprises ne disposent pas d'un inventaire complet de leurs API**.

- **Cloudflare a identifié près de 31 % de points de terminaison d'API REST *supplémentaires* via l'apprentissage automatique (Machine Learning)** que par l'intermédiaire d'identifiants de session fournis par les clients.[2]
- ***Plus de 15 000 comptes*** utilisateurs de Cloudflare ont vu des points de terminaison d'API identifiés uniquement par le biais de méthodes reposant sur le Machine Learning.[7]

Les API non gérées ou sécurisées par l'entreprise qui les utilisent (un phénomène également connu sous le terme d'API « fantômes » ou Shadow API) sont souvent introduites par des développeurs ou des utilisateurs particuliers cherchant à exécuter des fonctions métier spécifiques.

Si elles ne sont pas intrinsèquement malveillantes, **les API fantômes constituent par essence des surfaces d'attaque non protégées, source de nouveaux risques**.

Exploitées par des acteurs malveillants, ces API fantômes peuvent conduire à une exposition de données, à des vulnérabilités non corrigées, à des violations de la conformité des données, à des mouvements latéraux et à bien d'autres menaces.

CONTRÔLE DE LA VISIBILITÉ

Comment identifiez-vous et cataloguez-vous vos API à l'heure actuelle ?

L'inventaire d'API d'une entreprise ou d'un développeur est compilé par l'intermédiaire de schémas d'API, c'est-à-dire les métadonnées qui définissent les spécifications entourant la validité des requêtes et des réponses liées aux API. Ces schémas d'API (souvent documentés à l'aide de spécifications OpenAPI) comprennent l'hôte de l'API, la méthode HTTP, le chemin et les autres conditions établies par les développeurs, comme les variables de chemin et de requête.

Le risque lié aux API fantômes

Cloudflare constate souvent que les entreprises situées à un stade précoce de leur parcours de gestion des API emploient une approche « e-mail et demande de liste » (email and ask). Cette dernière dresse alors un inventaire ponctuel susceptible d'évoluer avec la version de code suivante. Toutefois, cette approche manuelle repose généralement sur une « connaissance tribale », sujette aux erreurs manuelles.

Imaginons que l'équipe informatique d'une entreprise de soins de santé n'ait pas conscience qu'une API accorde l'accès à certains systèmes à ses fournisseurs. Si un fournisseur se retrouve compromis, un acteur malveillant pourrait abuser de l'API pour exfiltrer des données médicales sur les patients.

La violation de données survenue en 2019 chez Quest Diagnostics, par exemple, a exposé les données de près de 12 millions de patients, lorsqu'un utilisateur non autorisé a obtenu l'accès à une API qui envoyait des informations à des partenaires de facturation.

En 2022, le fournisseur de télécommunications australien Optus a fait l'objet d'une violation, censément due à un acteur malveillant ayant réussi à accéder à sa base de données client via une API non authentifiée.

La croissance de l'économie autour des API entraîne également une augmentation des problèmes de perte de contrôle et un accroissement de la complexité liée au développement, à la gestion et à la sécurité des API.

CONTRÔLE DE LA SÉCURITÉ

Comment surveillez-vous la manière dont vos API autorisent les accès en « écriture » ?

Une fois les données agrégées sur l'ensemble des API des comptes, Cloudflare a découvert que **59,2 % des entreprises autorisaient les accès « en écriture » à au moins la moitié de leurs API.**[8]

Les API en accès « lecture uniquement » (GET) extraient et ingèrent les informations à partir d'un système. En revanche, les API disposant d'un accès « en écriture » (POST, PUT, DELETE) permettent également aux utilisateurs et aux autres applications d'effectuer des mises à jour (modifications) sur un système.

De nombreuses violations d'API surviennent en raison d'autorisations permissives, lorsque les utilisateurs se voient accorder un trop grand nombre de privilèges ou un accès aux données d'autres utilisateurs. Une API qui accorde un accès « en écriture » à la mauvaise personne peut ouvrir la voie à une attaque, comme celles décrites dans ce rapport.

Erreurs courantes relatives aux API

Une fois qu'une entreprise a identifié avec précision (puis enregistré ou supprimé) les points de terminaison d'API, elle doit savoir lesquelles fonctionnent correctement et lesquelles connaissent des dysfonctionnements. Les erreurs relatives aux API peuvent signaler la présence d'une cyberattaque ou de problèmes de performances au niveau des applications et ces anomalies peuvent, à leur tour, empêcher le fonctionnement légitime de l'entreprise.

Les codes de statut HTTP sont des codes à 3 chiffres, généralement utilisés pour indiquer si une application fonctionne correctement ou si cette dernière rencontre une erreur.

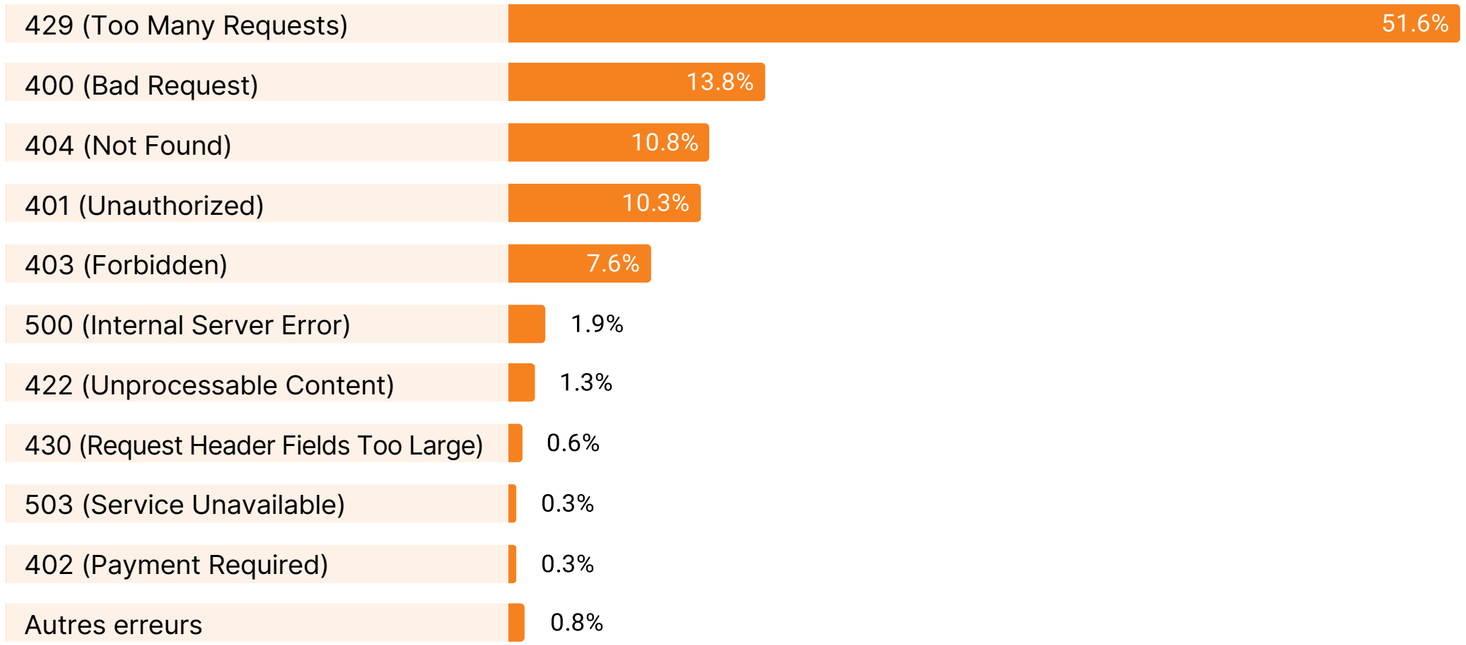
Pour les API et les autres requêtes HTTP, un code de statut commençant par « 2 » (codes de succès 2xx) indique que l'action d'un client a bien été reçue, comprise et acceptée (c'est-à-dire, la réussite de cette action).

Toutefois, lorsque les utilisateurs d'une application ne parviennent pas à joindre la destination souhaitée, ils peuvent, à la place, être redirigés (redirection 3xx) ou rencontrer des erreurs du client 4xx ou des erreurs côté serveur 5xx.

Cloudflare a observé des milliers de milliards d'erreurs de trafic provenant des origines des API, dont **plus de la moitié (51,6 %) renvoyaient un « code 429 » : « Trop de requêtes ».**[3]

Également connue sous le terme de « contrôle du volume de requêtes », l'erreur 429 se produit lorsque le client a envoyé trop de requêtes pendant la période spécifiée par le serveur.

Erreurs courantes relatives aux API



Référez-vous à l'annexe pour la liste des descriptions des erreurs.

Le risque lié au mauvais diagnostic des erreurs concernant les API

Une **erreur 429 (l'erreur relative aux API la plus fréquente**, comme montré ci-dessus) indique que le serveur a automatiquement jugulé le trafic de l'API lorsqu'une certaine action a été entreprise (comme le fait qu'une adresse IP donnée dépasse un certain nombre de requêtes par minute vers un point de terminaison `/login`).

Or, si une entreprise a déployé des mesures de contrôle du volume de requêtes définies manuellement (plutôt qu'un contrôle adaptatif), ces dernières peuvent devenir rapidement obsolètes. Et si le point de terminaison `/login` rencontrait un trafic supérieur à la moyenne en raison du succès d'une campagne marketing plutôt que du fait d'une attaque ? Dans ce scénario, le contrôle manuel du volume de requêtes pourrait empêcher les transactions légitimes.

Un autre exemple d'erreur dont la cause est souvent « mal diagnostiquée » est **l'erreur 401 « Unauthorized » (Utilisateur non authentifié, la quatrième erreur la plus courante observée par Cloudflare au sein du trafic lié aux API)**.

Une erreur 401 indique que les identifiants de l'utilisateur n'existent pas ou qu'ils ne contiennent pas le niveau d'accès approprié pour la ressource demandée. Toutefois, comme pour les autres codes d'erreur HTTP, le code peut résulter d'une menace (comme une tentative d'attaque Broken Object Level Authorization, c'est-à-dire une violation de l'autorisation au niveau de l'objet susceptible d'entraîner une prise de contrôle totale du compte) ou simplement être dû à la mauvaise saisie de ses identifiants par un utilisateur légitime.

Nouvel exemple de « mauvais diagnostic » du trafic lié aux API, au début de l'année 2023, Google a mis en garde les propriétaires de sites web et certains réseaux de diffusion de contenu contre l'utilisation de mauvais codes de statut dans le but de limiter la vitesse d'exploration du Googlebot (légitime).

Comme Google l'a rappelé aux utilisateurs, *« les erreurs du client ne reflètent rien de moins que ce que leur nom indique : des erreurs du client… Elles indiquent simplement que la requête du client ne s'est pas déroulée correctement d'une manière ou d'une autre »*.

CONTRÔLE DES PERFORMANCES

Comment surveillez-vous et évaluez-vous les erreurs relatives aux API ?

Toutes les erreurs relatives aux API ne sont pas forcément dues à des attaques. La bonne compréhension de la cause première de ces erreurs (et des tendances derrière ces problèmes) nécessite une journalisation constante du trafic lié aux API et une analyse des tendances au fil du temps.

Savez-vous quelle quantité de votre trafic d'API fait l'objet d'un contrôle du volume des requêtes ? Quelle quantité se voit interdire l'accès (en raison de mauvaises autorisations) ? Pouvez-vous confirmer que les erreurs sont dues à des attaques plutôt qu'à des identifiants utilisateur expirés (ou saisis de manière incorrecte) ?

Principales vulnérabilités en matière de sécurité des API

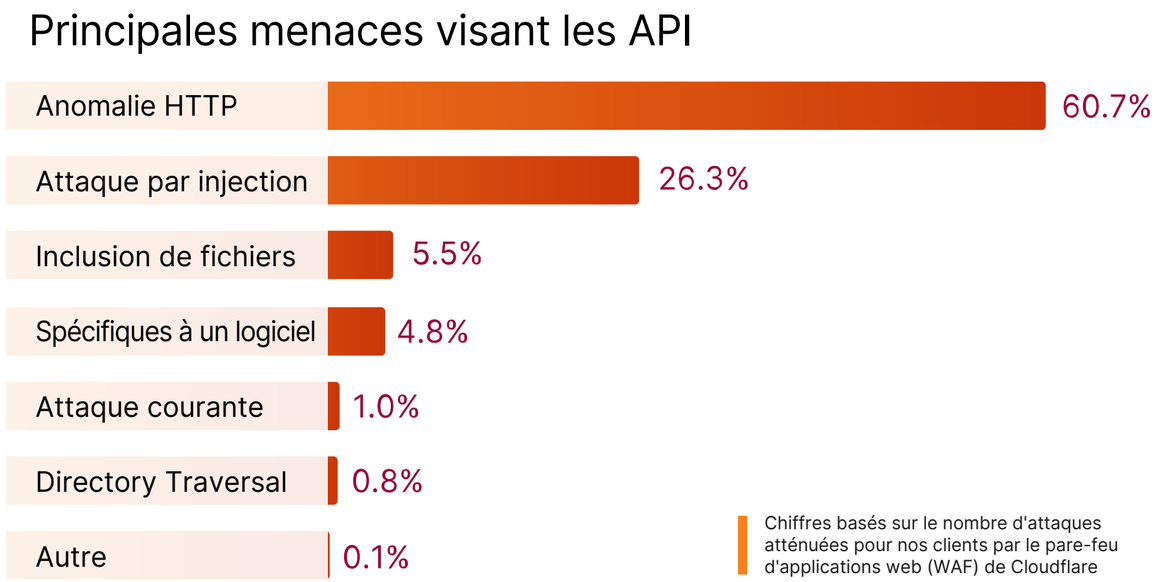
Les API sont difficiles à protéger contre les abus. Elles exigent un haut degré d'approfondissement en matière de contexte opérationnel, de méthodes d'identification et de mesures de vérification des accès par rapport aux autres services de sécurité des applications web.

Considérez, par exemple, les caractéristiques suivantes :

Une application web	Une API
Est **visible** de l'utilisateur final.	Est **invisible** à l'utilisateur d'une application.
Est **accessible aux utilisateurs finaux** par l'intermédiaire d'un navigateur web.	Permet **aux systèmes et aux applications** d'échanger des données.
Visualise les données du back-end (en utilisant le HTML, le CSS et le JavaScript) à l'aide de diverses interactions utilisateur.	**Transporte les données** en accédant aux serveurs et aux applications à l'aide d'un format défini (le plus couramment, RESTful JSON, gRPC, XML ou GraphQL).
Est généralement protégée à l'aide d'un modèle de **« sécurité négative »** qui bloque le trafic malveillant connu.	Est plus efficacement protégée par l'intermédiaire d'un modèle de **« sécurité positive »** n'autorisant que le trafic validé et authentifié.

Pour ces raisons (et d'autres), la surveillance régulière et automatisée des API est essentielle à l'identification rapide des menaces et au traitement de ces dernières.

Vous trouverez ci-dessous un instantané des menaces les plus fréquentes envers les API que Cloudflare a atténuées pour ses clients en 2023[9] :



Vous trouverez une description plus détaillée de ces types d'attaques en annexe.

Le rôle des vulnérabilités des API lors d'une attaque MDM

La gestion des appareils mobiles (Mobile Device Management, MDM) aide les entreprises à gérer l'ensemble de leurs appareils géographiquement dispersés à partir d'une plateforme unique. Grâce à la MDM, les équipes informatiques peuvent déployer et contrôler les applications sur les appareils gérés à l'aide d'API intégrées à ces derniers.

La commodité et la simplicité des systèmes MDM doivent toutefois être pondérées au regard des risques. Ces systèmes constituent en effet des cibles privilégiées, car elles peuvent offrir aux pirates un accès de haut niveau à des milliers d'appareils mobiles.

En août 2023, la Cybersecurity and Infrastructure Security Agency (CISA, l'agence de cybersécurité et de sécurité des infrastructures) et le Norwegian National Cyber Security Centre (NCSC-NO, le centre de cybersécurité national norvégien) ont émis un avis consultatif conjoint sur la cybersécurité prévenant le public du fait que les acteurs malveillants enchaînaient deux vulnérabilités pour **exploiter l'offre Endpoint Manager Mobile (EPMM) d'Ivanti, anciennement connue sous le nom de MobileIron Core**.

Ces derniers utilisaient plusieurs méthodes, comme les techniques MITRE ATT&CK® décrites dans le tableau suivant. L'adhésion aux cadres tels que le MITRE ATT&CK et le Top 10 de l'OWASP en matière de sécurité des API permet de proposer une base solide à une cybersécurité plus résiliente, comprenant notamment des mesures de défense des API plus robustes.

Exemple de technique *(pour la liste complète, cliquez ici)*	**Utilisation**
Exploitation d'application en contact avec le public	Les acteurs malveillants ont tiré parti de la vulnérabilité CVE-2023-35078 au sein des équipements EPMM d'Ivanti depuis au moins avril 2023.
Interpréteur de commandes et de scripts	Les acteurs malveillants pourraient avoir exploité la vulnérabilité CVE-2023-35081 pour importer des webshells sur l'appareil EPMM et exécuter des commandes.
Identification de compte : compte de domaine	Les pirates ont exploité la vulnérabilité CVE-2023-35078 pour dresser une liste des utilisateurs et des administrateurs d'un appareil EPMM. **Dans ce scénario, ils se sont servis du chemin d'API** `/mifs/aad/api/v2/authorized/users` **pour lister les utilisateurs et les administrateurs de l'appareil EPMM**.
Identification de système distant	Les pirates ont récupéré des points de terminaison LDAP.
Composant logiciel de serveur : webshell	Les acteurs malveillants ont implanté des webshells au sein de l'infrastructure compromise.
Proxy	Les acteurs malveillants ont tiré parti de routeurs SOHO compromis pour compromettre l'infrastructure et se placer derrière elle en tant que proxy.

Deux moyens d'atténuer les vulnérabilités courantes affectant les API

1. Validation de schéma

Les anomalies HTTP, comme l'absence d'agents utilisateur (le logiciel qui récupère le contenu Internet pour les utilisateurs finaux), les noms de méthode mal formés, les ports non standard et bien d'autres, sont des signaux courants de requêtes malveillantes. Or, comme montré plus haut, ces types d'anomalies HTTP composaient la majorité des menaces visant les API atténuées par Cloudflare.

La validation de schéma constitue un moyen utile d'identifier les anomalies HTTP afin de n'autoriser que le trafic « propre » de chaque API vers vos serveurs d'API. Le schéma d'API définit la validité des requêtes d'API en fonction de plusieurs propriétés des requêtes, comme le point de terminaison cible, le format de variable du chemin ou de la requête et la méthode HTTP.

2. S'attaquer aux lacunes de l'authentification

L'absence d'authentification (ou le fait que cette dernière ne fonctionne pas) au sein des API publiques constitue un autre problème grave, comme l'ont montré les gros titres faisant état de violations de données en lien avec les API.

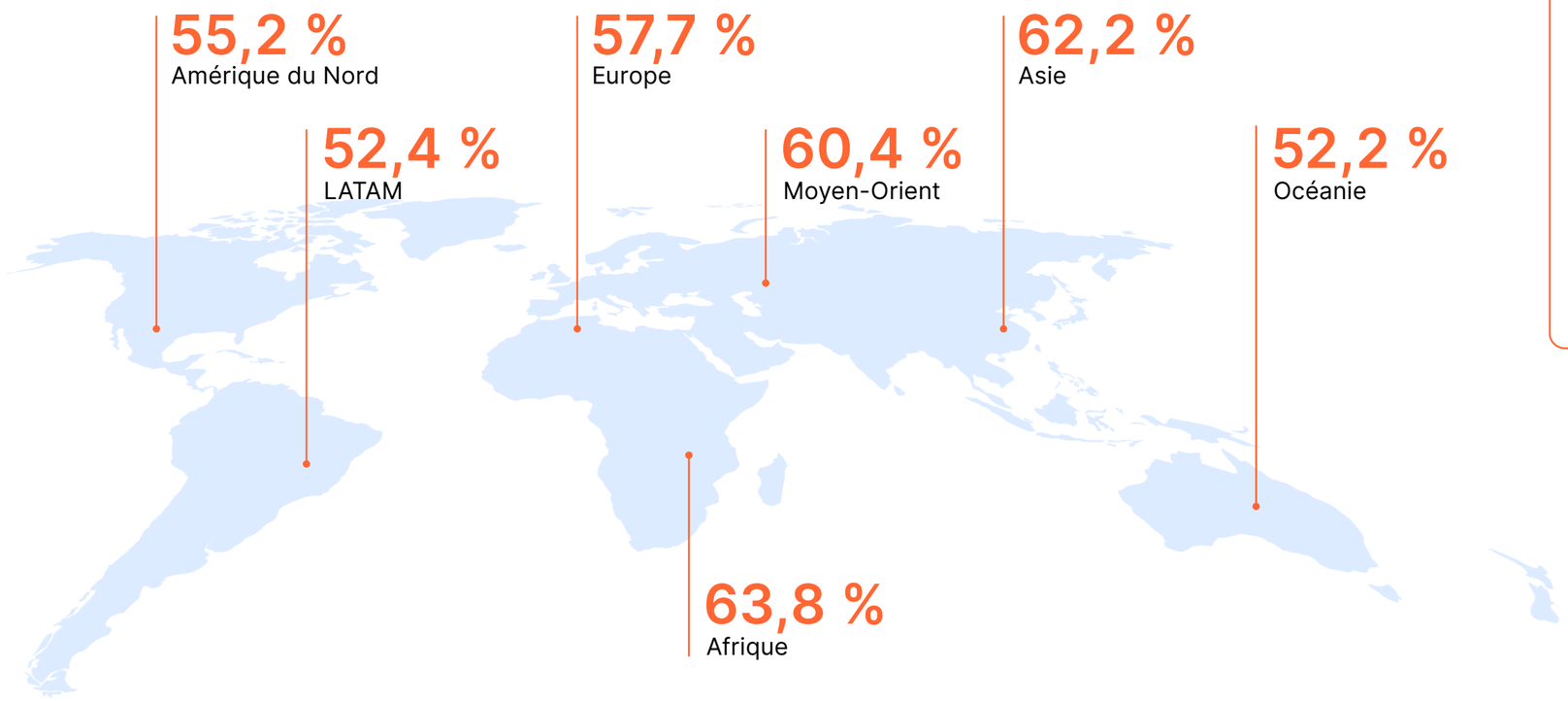
Vous trouverez ci-dessous quatre moyens de s'attaquer aux lacunes du processus d'authentification susceptibles d'exposer des données sensibles par l'intermédiaire de vos API :

- Pour commencer, appliquez une authentification à chaque API publiquement accessible, sauf exception approuvée par l'activité.
- Limitez la vitesse des requêtes d'API vers vos serveurs afin de ralentir les pirates potentiels.
- Bloquez les volumes anormaux de flux de données sensibles sortants.
- Empêchez les acteurs malveillants d'ignorer les séquences légitimes de requêtes d'API.

Le monde orienté API

Tendances régionales

Dans chaque région protégée par Cloudflare, le trafic lié aux API représentait plus de la moitié du trafic HTTP dynamique de cette dernière[10] :



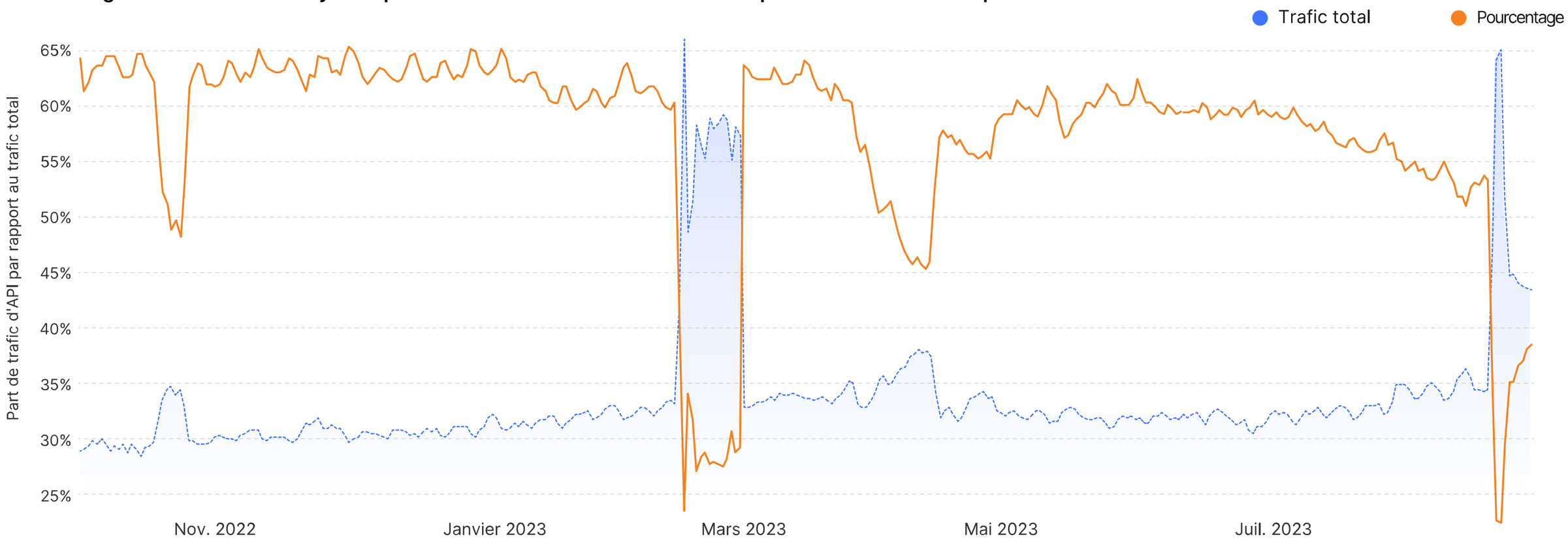
Dans l'ensemble, le trafic d'API total a augmenté régulièrement tout au long de l'année 2023. Nous avons toutefois constaté des fluctuations remarquables dans les régions suivantes :

- Dans la région **LATAM**, le trafic lié aux API représentait **entre 46,1 % et 58,6 %** du trafic HTTP dynamique.
- Dans la région **Océanie**, le trafic lié aux API représentait **entre 44,1 % et 57,4 %** du trafic HTTP dynamique.
- Enfin, c'est dans la région **Moyen-Orient** que **le trafic lié aux API a le plus varié**, comme nous le verrons plus en détail dans la section suivante.

Pics de trafic au Moyen-Orient

Les fortes fluctuations du trafic en lien avec les API au Moyen-Orient ont coïncidé avec une flambée soudaine et temporaire du **trafic général vers un outil d'anonymisation** connu pour aider à circonvenir les restrictions du réseau. Cloudflare a observé que les pics de trafic survenus en 2023 vers cet outil se sont produits peu de temps après les coupures d'Internet d'origine gouvernementale.

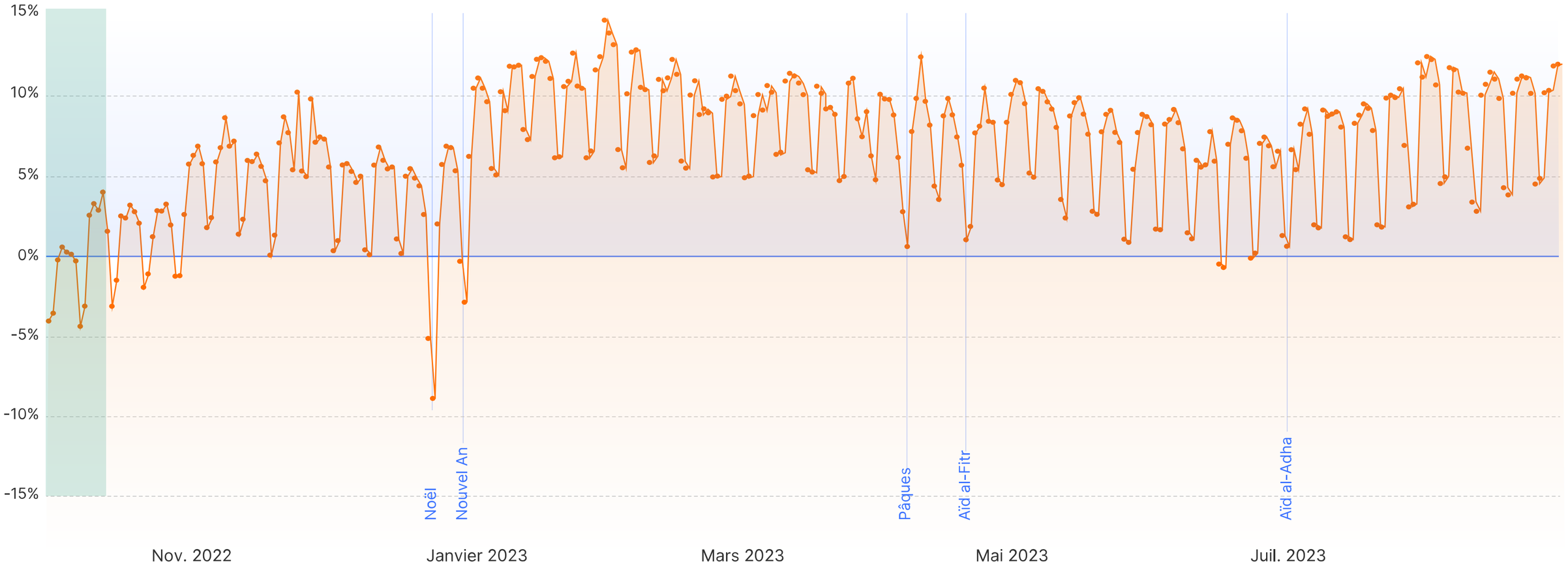
Pourcentage de trafic de cache dynamique lié aux API et assorti d'un code de réponse 200 au fil du temps



Le trafic lié aux API ralentit-il l'ensemble ?

Si le trafic lié aux API est souvent considéré comme des conversations entre bots, les données de Cloudflare ont révélé des fluctuations claires (pics et baisses) de ce trafic tout au long de l'année, notamment autour des principaux jours fériés.

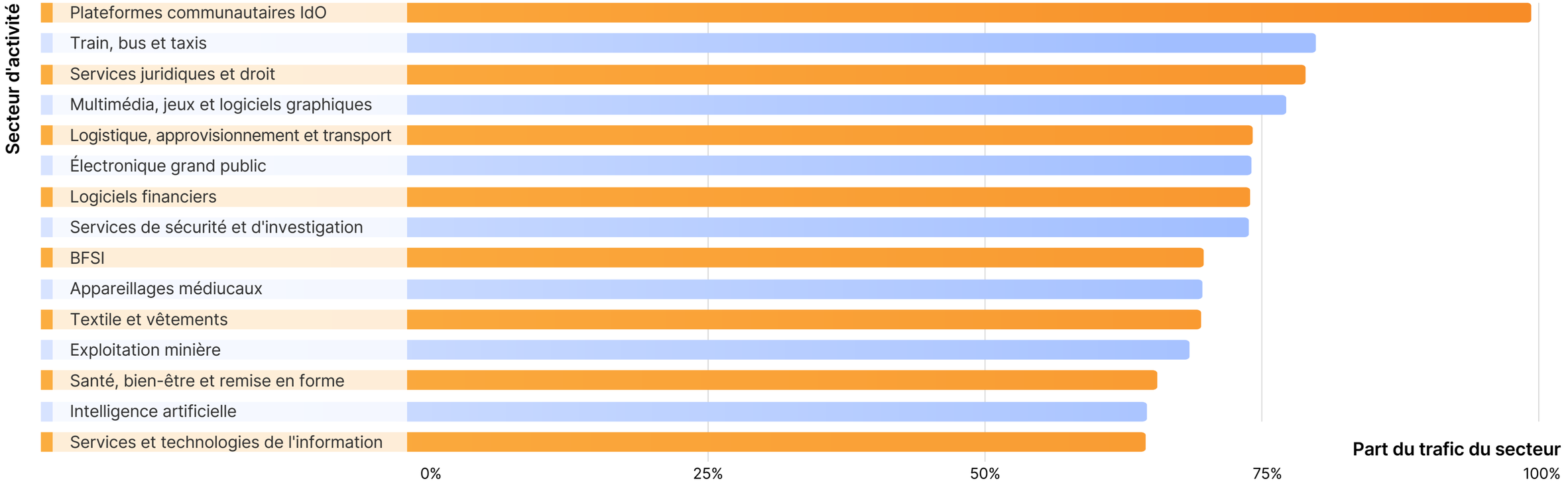
Il semble que lorsque les utilisateurs avaient le plus de chances d'être hors ligne, par exemple, **le 25 décembre (Noël), le 9 avril (Pâques) ou le 22 avril (Aïd al-Fitr), le trafic en lien avec les API ait décliné de manière notable.**[11]



Le trafic lié aux API en fonction des secteurs

En plus des variations d'ordre géographique, **certains secteurs ont fait état d'une plus grande part de trafic lié aux API que d'autres**.

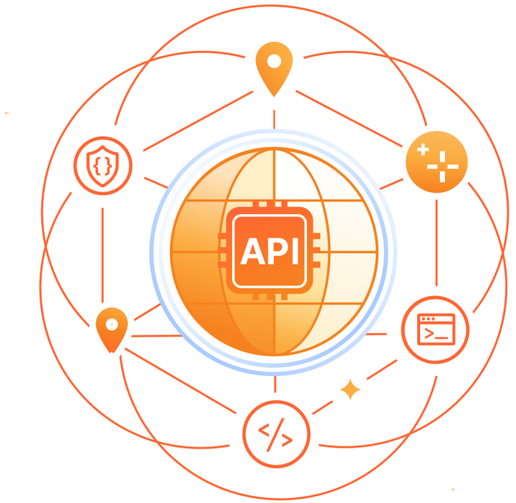
Les 15 principaux secteurs dans lesquels Cloudflare a observé un volume plus élevé de trafic orienté API (par rapport au trafic HTTP dynamique total du secteur)[12] étaient les suivants :



Indicateurs sectoriels

Au lieu de développer de nouvelles fonctions à partir de zéro, une application, un site web ou une application mobile peut enrichir l'expérience utilisateur en ajoutant de nouvelles fonctionnalités par le biais d'API.

Par exemple, plutôt que de créer ses propres services de paiement depuis le départ, une application de covoiturage peut ajouter le paiement via les API d'entreprises de paiement. Les API des commerçants en ligne peuvent personnaliser l'expérience client en proposant des cabines d'essayage virtuelles, des recommandations de produits et le suivi des commandes.



Les API sont utiles dans n'importe quel secteur, à n'importe quel niveau.

Voici les secteurs qui détiennent la part la plus élevée de trafic lié aux API dans chaque région spécifique[12] :

Afrique

1. **Services pour installations**
2. Exploitation minière
3. Marchés de capitaux
4. Collecte de fonds
5. Cartes de paiement et traitement des transactions

Asie

1. **Plateformes communautaires IdO**
2. Exploitation minière
3. Textile et vêtements
4. Banque, assurances et services financiers (BFSI)
5. Intelligence artificielle

Europe

1. **Multimédia, jeux et logiciels graphiques**
2. Logiciels d'élaboration de contenus et collaboratifs
3. Équipements médicaux
4. Textile et vêtements
5. Services juridiques

Amérique latine

1. **Exploitation minière**
2. Logiciels financiers
3. Multimédia, jeux et logiciels graphiques
4. Marchés de capitaux
5. Pratique juridique

Moyen-Orient

1. **Collecte de fonds**
2. Services juridiques
3. Connectivité sans fil
4. Marchés de capitaux
5. Transport routier/ ferroviaire

Moyen-Orient

1. **Services juridiques**
2. Trains, bus et taxis
3. Électronique grand public
4. Services de sécurité et d'investigation
5. Logistique, approvisionnement et transport

Amérique du Nord

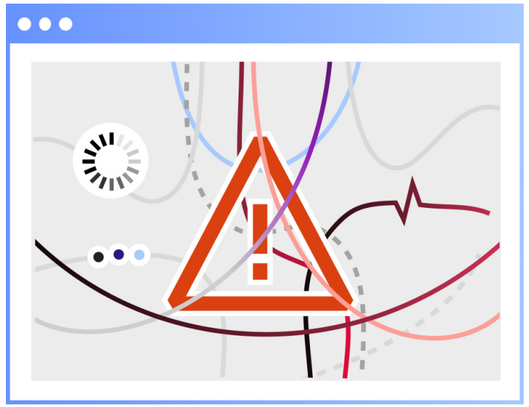
1. **Exploitation minière**
2. Textile et vêtements
3. Marchés de capitaux
4. Services de sécurité et d'investigation
5. Produits pharmaceutiques, biotechnologies et santé

Prévisions pour 2024 et au-delà

Plus les consommateurs et les utilisateurs finaux continueront de s'attendre à des expériences web et mobiles toujours plus dynamiques, plus les équipes de développement et les équipes chargées des API subiront des pressions concernant le déploiement et la maintenance d'un nombre d'API croissant. Les développeurs d'applications bien intentionnés continueront de déployer des API à un rythme rapide, parfois sans consulter les autres parties prenantes en matière d'informatique et de sécurité.

Ce manque de cohésion au niveau de l'approche poussera les entreprises dans leurs derniers retranchements lorsqu'elles devront faire face aux défis suivants :

1 Accélération de la perte de contrôle et de la complexité



Les décisionnaires en matière d'IT déclarent que le premier facteur contribuant à la perte de contrôle sur les environnements informatiques et la sécurité réside dans le « nombre global d'applications », suivi par « l'augmentation des emplacements d'applications ».

Pourtant, dans la plupart des entreprises, ces équipes demeurent cloisonnées :

73 % des développeurs affirment que les tâches ou les outils que leur équipe de sécurité les oblige à utiliser « interfèrent avec leur productivité et leur capacité d'innovation ».

87 % des DSI pensent que les ingénieurs logiciels et les développeurs « font des compromis sur les politiques de sécurité et les mesures de contrôle pour lancer de nouveaux produits et services plus rapidement sur le marché ».

<50 % des RSSI ont l'impression que les développeurs sont « très familiers » des risques envers la sécurité liés aux outils de développement et d'automatisation.

Les équipes chargées de l'informatique, de la sécurité et du développement d'applications partagent toutes la responsabilité de la protection de l'immense surface d'attaque impliquée par la présence de milliers de ressources soutenues par des API.

À moins qu'elles ne corrigent l'écart entre l'IT, la sécurité et le développement d'applications grâce à la protection automatisée des API, **les entreprises peuvent s'attendre à une hausse des risques liés aux API et de la complexité de gestion de ces dernières**.

2 Un accès facilité à l'IA menant à davantage de risques visant les API

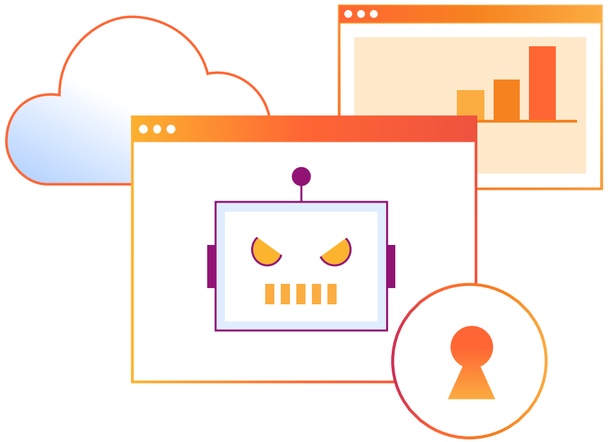
Les analystes prédisent que d'ici 2026, plus de 80 % des entreprises auront utilisé des API ou des modèles d'intelligence artificielle générative (IAG) et/ou déployé des applications IAG (comme ChatGPT) dans les environnements de production.

En l'absence de front-end par application web, les utilisateurs accèdent généralement aux modèles d'IAG soit directement dans le cadre d'une fonction interne, soit par l'intermédiaire d'autres applications et d'autres utilisateurs via des API publiques, comme les API ChatGPT et Whisper d'OpenAI. **Comme l'utilisation de l'IAG augmente l'utilisation des API de manière spectaculaire, les services d'IAG attireront également davantage d'attaques à l'encontre de leurs API**.

Pour prendre un exemple, des concurrents ou des pirates « appelant » l'API d'un produit des millions de fois afin d'extraire et de dérober des données auraient un coût direct relativement négligeable sur la facture d'infrastructure de la victime. En revanche, un acteur malveillant tirant parti des modèles génératifs de la victime par l'intermédiaire d'API coûterait davantage (plusieurs centimes par appel). Un tel acteur malveillant effectuant des millions d'appels à l'API d'une application IA entraînerait une perte financière immédiate.

En outre, et ce même lorsque l'IAG est utilisée à des fins bienveillantes, il s'agit toujours d'un territoire inconnu (c.-à-d. risqué) pour de nombreux développeurs. Forrester prédit qu'en 2024, sans les garde-fous appropriés, « au moins trois violations de données seront publiquement imputées à du code non sécurisé généré par IA, que ce soit du fait de failles de sécurité dans le code généré lui-même ou de vulnérabilités dans les dépendances suggérées par l'IA ».

3 Augmentation des fraudes basées sur la logique métier



Les années 2020 ont vu les opérateurs de bots cibler les applications web utilisant des versions de navigateurs web instrumentés afin de créer des bots sophistiqués basés sur navigateur. En parallèle, la plupart des applications modernes font usage d'API en arrière-plan afin de finaliser les actions des utilisateurs, comme la création de compte, la connexion, le renseignement de formulaires et les procédures de transaction monétaire.

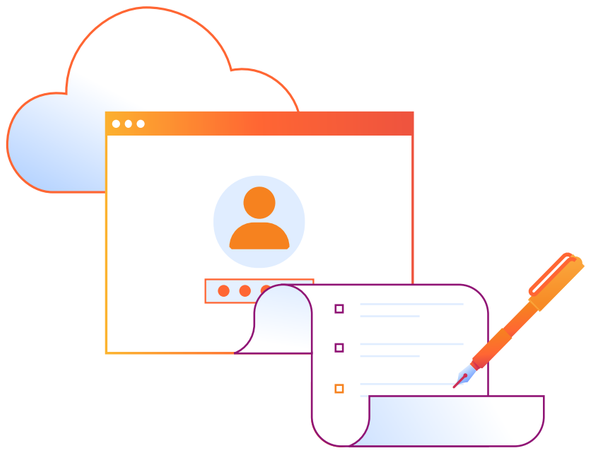
Pour l'année 2024, nous nous attendons à ce que les opérateurs de bots attaquent davantage d'API à l'œuvre derrière ces flux de travail directement, car ces attaques sont plus efficaces (les API ont tendance à changer moins souvent que les interfaces des applications web) et font face à moins de protections (par rapport aux applications web).

Prenons l'exemple d'une création de faux compte dans le secteur des paris sportifs et des ligues sportives virtuelles (« fantasy leagues »). Un utilisateur disposant de plusieurs comptes pour placer différents paris et déclarer plusieurs compositions d'équipes dispose de chances accrues de gagner, qui se traduisent souvent par des gains monétaires. L'idée d'automatiser la création de nouveaux comptes à grande échelle est donc d'autant plus lucrative.

Des motivations similaires interviennent dans le cas des attaques par bourrage d'identifiants (visant des cibles au niveau desquelles aucune authentification multifactorielle n'a été mise en place ou, à défaut, des mesures d'authentification faciles à contourner), mais aussi des achats frauduleux d'articles en quantité limitée.

Dans ce type de situation, les entreprises ont besoin d'un service d'informations basé sur la logique métier au sein de leurs outils de sécurité des API. Par exemple, pour identifier les séquences anormales envoyées par les acteurs malveillants afin de soutenir leurs tentatives de fraude, mais aussi pour identifier à quel moment l'appel d'API présente des caractéristiques comportementales anormales, comme le fait d'essayer de finaliser les transactions plus rapidement que le volume de transactions de référence pour cette API.

4 Inflation de la réglementation et de la gouvernance



Les entreprises doivent également s'attendre à **davantage de gouvernance et d'initiatives visant à réguler la sécurité et la confidentialité des API**.

Pour prendre un exemple, la norme PCI DSS (« Payment Card Industry Data Security Standard », norme de sécurité de l'industrie des cartes de paiement) est un cadre de travail conçu pour guider les entreprises dans les processus de gestion des transactions des titulaires de cartes et des données d'authentification de paiement. **Les nouvelles conditions PCI DSS v4.0 (la première version à aborder explicitement la sécurité des API) prendront effet le 31 mars 2024**.

Avec l'entrée en vigueur de la norme PCI DSS v4.0, toutes les entreprises qui transmettent ou traitent des paiements par carte devront remédier aux vulnérabilités des API et veiller à mettre en place un processus d'authentification approprié pour ces dernières, parmi bien d'autres conditions. Tout manquement au respect des exigences PCI DSS pourra être passible de lourdes amendes et d'autres pénalités.

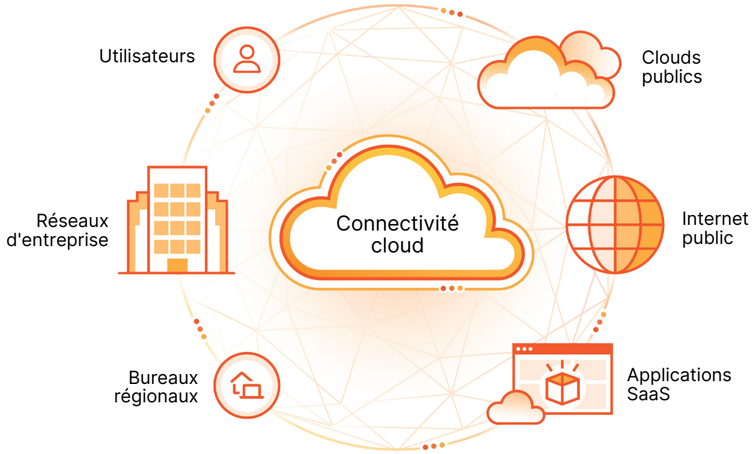
Autre exemple hautement régulé, le secteur de la santé peut s'attendre à davantage de surveillance autour des API, compte tenu de leur capacité à transmettre des informations médicales électroniques protégées (electronic Protected Health Information, ePHI) entre systèmes.

En juillet 2023, la Federal Trade Commission (Commission fédérale du commerce) et le département des services sociaux de l'Office for Civil Rights (OCR, le Bureau des droits civiques) des États-Unis ont intensifié leurs efforts de surveillance des risques envers la confidentialité des applications de santé. Ces institutions ont également averti les parties concernées de pénalités financières pour tout manquement aux obligations de divulgation en cas de violation des données médicales personnelles.

Recommandations

Comme pour n'importe quel équipement logiciel, les API connaîtront des vulnérabilités. Si personne ne peut empêcher les acteurs malveillants d'essayer constamment de nouvelles techniques pour percer la sécurité des applications et des API, les entreprises peuvent identifier, protéger et gérer ces dernières à l'aide d'une approche globale intégrant les bonnes pratiques suivantes :

1 Unifier la gestion du développement, de la visibilité, des performances et de la sécurité des applications grâce à la connectivité cloud



Dans de nombreuses entreprises, l'infrastructure propriétaire et les besoins uniques en matière de conformité (de même que les problèmes de semi-compatibilité entre les processus et les configurations) compliquent la connexion entre les clouds, les applications SaaS, les applications web et l'infrastructure sur site. Ces domaines n'ont tout simplement pas été conçus pour fonctionner ensemble de manière simple et sécurisée.

La connectivité cloud est une nouvelle approche permettant de fournir les nombreux services dont les entreprises ont besoin pour sécuriser et connecter leurs environnements numériques. Elle prend la forme d'une plateforme intelligente de services cloud-native et programmables autorisant une connectivité point à point (any-to-any) entre les réseaux, les environnements cloud, les applications et les utilisateurs.

La connectivité cloud assure le tissu conjonctif entre les services de déploiement d'applications et de défense en profondeur des API, en proposant notamment les suivants :

- **Un service automatisé d'identification des API et de visibilité** offrant aux entreprises un inventaire clair de leur parc d'API.
- **Des processus d'authentification et d'autorisation modernes** intégrés dès le départ.
- **Un service de gestion des points de terminaison d'API** conçu pour surveiller divers indicateurs, comme la latence, les erreurs, le taux d'erreurs et la taille des réponses pour les domaines orientés API.
- **Des protections de couche 7 (L7) pour les API**, incluant notamment le contrôle du volume de requêtes et une protection contre les attaques par déni de service, les tentatives de connexion par force brute et les autres abus touchant les API.
- **Un service de détection des vulnérabilités zero-day** (les nouvelles vulnérabilités découvertes dans les logiciels et qui ne disposent pas d'un correctif) conçu pour détecter les zero-day avant qu'une attaque ne survienne.

2 Migrer vers un modèle de « sécurité positive » à l'aide d'une passerelle d'API

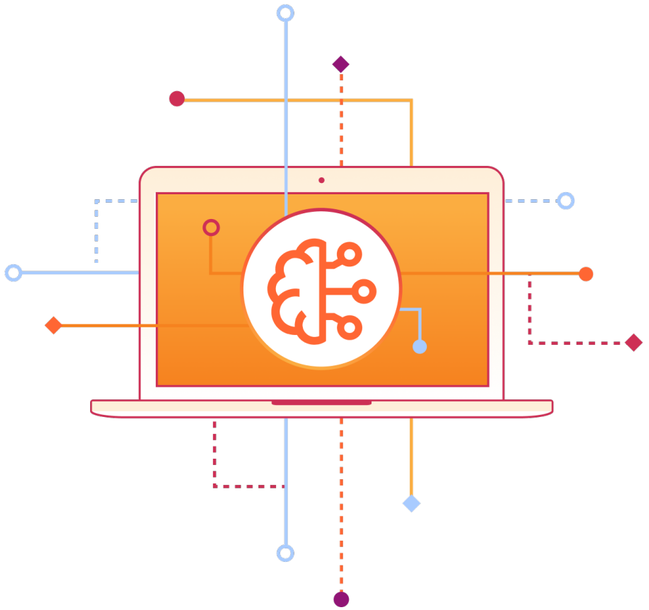
Nous estimons le nombre d'API publiques et privées en cours d'utilisation à 200 millions. Or, ce chiffre en croissance constante implique que les responsables en matière d'IT et de sécurité ne peuvent « suivre le rythme » de manière réaliste concernant les performances, le comportement et l'exposition aux risques de chaque API.

Traditionnellement, les applications web sont protégées à l'aide d'un modèle de « sécurité négative » appliqué par un pare-feu d'applications web (WAF) autorisant toutes les transmissions, à l'exception des requêtes provenant d'adresses IP, d'ASN et de pays « problématiques » ou les requêtes comportant des signatures également problématiques (comme les tentatives SQLi). Cette situation est due au fait que les utilisateurs peuvent accéder aux applications web et interagir avec celles-ci d'un certain nombre de manières. Dans ce type de modèle, le pare-feu WAF bloquera les requêtes « connues pour être malveillantes », tout en autorisant le reste du trafic.

A contrario, un modèle de « sécurité positive » se révèle plus approprié pour les API, car ces dernières disposent d'un format structuré régissant les interactions avec elles. S'agissant de l'inverse d'une approche de sécurité négative, **le modèle de sécurité positive n'autorise que les identités et les comportements « connus pour être fiables » (la « fiabilité » étant définie par le schéma d'API)**, tout en rejetant toutes les autres formes de trafic.

Les entreprises utilisant un modèle de sécurité positive protègent leurs API en acceptant uniquement le trafic correspondant à leurs schémas. Dans les faits, elles peuvent ainsi bloquer plus efficacement les requêtes mal formées et les anomalies HTTP, comme les attaques par bourrage d'identifiants (Credential Stuffing) et les outils d'analyse automatisés.

3 Utiliser les technologies d'apprentissage automatique pour libérer des ressources et réduire les coûts

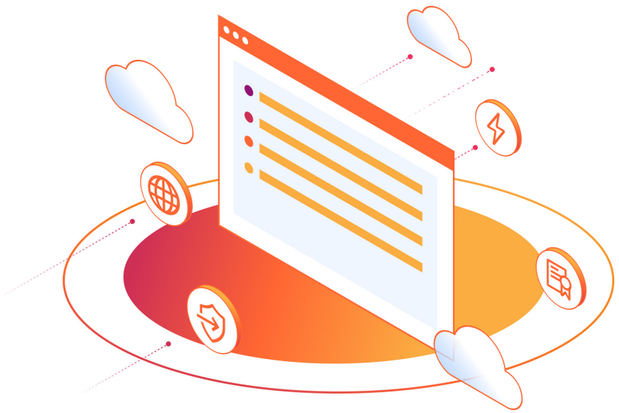


Sans automatisation ni outils spécifiquement conçus pour les API, les parties prenantes en matière d'IT et de sécurité n'ont aucune chance de suivre le rythme de leurs équipes chargées des API.

Les entreprises peuvent toutefois rendre la gestion de la sécurité et de la visibilité des API plus efficace en utilisant des services de sécurité basés sur l'apprentissage automatique (Machine Learning). Le Machine Learning permet ainsi de rapidement :

- **Révéler** l'ensemble du trafic des API (y compris des API non authentifiées) vers un domaine, indépendamment des données basées sur un identifiant de session.
- **Détecter** les variations des attaques RCE, XSS et SQLi visant les API.
- **Nourrir** un classificateur permettant de faire la distinction entre les divers types de trafic et de vecteurs d'attaque sur les API.
- **Faire la différence** entre les pics légitimes du trafic des utilisateurs d'applications et les pics de trafic potentiellement malveillant dû à des bots.

4 Mesurer et améliorer le niveau de maturité en matière d'API de votre entreprise au fil du temps



L'approche la plus complète en matière de protection des API consiste à mettre en œuvre une plateforme globale de protection des API et des applications web (Web Application and API Protection, WAAP). Toutefois, une entreprise qui commence tout juste à reconnaître l'exposition de ses API pourrait ne pas trouver ce déploiement immédiatement envisageable.

Toute évolution se doit néanmoins de commencer à un moment ou un autre. Une entreprise qui a compris qu'elle doit être protégée peut alors progresser vers un système de gestion et de sécurité des API plus complet.

Niveau 1 : visibilité

Les entreprises doivent tout d'abord surveiller et gérer de manière formelle l'ensemble de leurs points de terminaison d'API, y compris ceux de leurs éventuelles API fantômes. Nombre d'entre elles ne parviennent toutefois pas à identifier leurs API aussi vite que leurs développeurs les produisent. Lorsqu'elles trouvent effectivement des API, il s'avère particulièrement difficile pour elles de tracer avec précision un schéma unique pour chacun des points de terminaison d'API (qui peuvent potentiellement se compter en centaines).

Grâce à un service de visibilité sur les API, les entreprises peuvent à la fois identifier automatiquement les points de terminaison d'API, mais aussi identifier qui possède une API donnée et comment cette dernière doit être utilisée.

Niveau 2 : protection générale contre les attaques web

Les applications web et les API fonctionnent souvent de manière conjointe (un site web d'e-commerce peut, par exemple, utiliser une API pour traiter les paiements). La nature intrinsèquement mondiale d'Internet expose cependant les sites web et les autres applications aux attaques lancées depuis différents endroits, avec différents niveaux d'ampleur et de complexité.

Les services suivants (abordés plus en détail ici) sont des exemples de « mises minimums » en matière de protection directe des applications web et des API contre les attaques DoS et DDoS, le bourrage d'identifiants, les vulnérabilités zero-day et les autres types de menaces :

- **Les services d'atténuation des attaques DDoS** se placent entre un serveur et l'Internet public afin d'empêcher les pics de trafic malveillant de submerger le serveur.
- Un **pare-feu d'applications web (WAF)** filtre le trafic connu pour (ou suspecté de) tirer parti des vulnérabilités des applications web.
- **La gestion de la certification de chiffrement** aide à gérer les éléments principaux du processus de chiffrement SSL/TLS.
- **Le contrôle avancé du volume de requêtes** protège les points de terminaison contre les attaques DoS, les tentatives de connexion par force brute et les autres pics de trafic lié aux API, sans pénaliser les utilisateurs légitimes.

Niveau 3 : protection contre les attaques visant spécifiquement les API

Les outils tels que les pare-feu WAF et les mesures de protection contre les attaques DDoS sont essentiels à la sécurité web et à l'expérience des utilisateurs d'applications (humains). Ces services sont toutefois conçus pour protéger les applications, pas spécifiquement les API.

Plus une entreprise expose de services par le biais d'API, plus elle devrait renforcer la sécurité de ses applications web à l'aide de solutions de sécurité et de gestion des API spécifiquement développées à cet effet.

Une solution de sécurité des API avancée, reposant sur l'apprentissage automatique non supervisé, est capable de développer des bases de référence distinctes pour chaque API et de prédire l'intention des requêtes effectuées aux API (qu'elles soient légitimes ou malveillantes).

Les entreprises savent que la sécurité des API est une donnée nouvelle pour de nombreux membres de leurs équipes. La sécurité ne peut pas être poursuivie comme un but en soi, mais doit servir une amélioration des résultats opérationnels. Certains de ces avantages résident dans la capacité à proposer plus rapidement ses produits, à réduire le nombre de lacunes de sécurité au sein des API publiées, mais aussi à disposer d'équipes de sécurité plus efficaces et, en définitive, d'équipes plus productives, que ce soit en matière de développement ou d'API.

Protégez les API qui soutiennent votre activité

Soutenu par la connectivité cloud de Cloudflare, le catalogue de produits de protection des API et des applications web (Web Application and API Protection, WAAP) intègre des fonctionnalités de premier ordre conçues pour assurer la sécurité et la productivité des applications et des API, mais aussi bloquer les attaques, les bots et bien d'autres fonctions.

En savoir plus

sur l'identification des API, la protection contre les menaces du Top 10 de l'OWASP en matière de sécurité des API, le protocole Mutual TLS (mTLS) et la protection des API, sans compromettre l'innovation.

Glossaire de la sécurité des API

Appel d'API ou requête d'API : un message envoyé à un serveur pour demander à une API de fournir un service ou une information.

Identification des API : l'identification des API désigne le processus qui vise à cataloguer l'ensemble des API internes et tierces utilisées au sein d'une entreprise.

Point de terminaison d'API : l'endroit où les requêtes d'API (également connues sous le nom d'appels d'API) sont satisfaites. Un point de terminaison d'API est pratiquement toujours hébergé sur un serveur.

Trafic d'API : toute requête HTTP assortie d'un type de contenu de réponse au format `XML`, `JSON`, `gRPC` ou similaire. Lorsque le type de contenu de réponse n'est pas disponible, comme c'est le cas pour les réponses atténuées, c'est le type de contenu Accept équivalent (spécifié par l'agent utilisateur) qui est utilisé à la place. Dans ce dernier cas, le trafic d'API n'est pas entièrement comptabilisé, mais constitue néanmoins toujours une bonne représentation à des fins d'informations et de statistiques.

Trafic lié aux bots/trafic automatisé : toute requête HTTP identifiée comme générée par un bot selon le système de gestion des bots de Cloudflare.

Violation de l'autorisation au niveau de l'objet (Broken Object Level Authorization, BOLA) : ce terme désigne la manipulation d'identifiants d'objets au sein d'une requête afin d'obtenir un accès non autorisé à des données sensibles. Dans une attaque BOLA, les acteurs malveillants accèdent à des objets (données) auxquels ils ne devraient pas avoir accès, en modifiant simplement les identifiants.

Défaillance de l'authentification des utilisateurs (Broken User Authentication) : si l'authentification est mise en œuvre de manière incorrecte, les acteurs malveillants peuvent être en mesure de se faire passer pour des utilisateurs d'API, afin d'accéder à des données confidentielles.

Client : la partie effectuant la requête HTTP. Il s'agit généralement d'un utilisateur final accédant à un site à partir d'un navigateur, mais il peut également s'agir d'un client API ou de n'importe quelle partie sollicitant des ressources de ce site.

Traversée de répertoires : également connue sous le nom d'attaque Directory Traversal ou d'attaque par traversée de chemin, la traversée de répertoires a pour objectif d'accéder à des fichiers et des répertoires stockés hors du dossier racine d'un site web.

Attaque par déni de service distribué (DDoS) : une attaque DDoS constitue une tentative malveillante de perturber le trafic normal du serveur, du service ou du réseau ciblé en submergeant ce dernier ou son infrastructure environnante sous un flot de trafic Internet.

Inclusion de fichier : cette vulnérabilité permet à un acteur malveillant d'inclure un fichier dans l'application cible. La vulnérabilité survient du fait de l'utilisation d'entrées fournies par l'utilisateur sans validation appropriée.

Anomalie HTTP : les anomalies HTTP, comme les noms de méthodes mal formés, les caractères nuls, les ports non standard ou une longueur de contenu de zéro dans une requête POST, sont des indicateurs courants d'attaques atténuées par les règles WAF gérées de Cloudflare. Vous trouverez une description détaillée de divers exemples de règles relatives aux anomalies HTTP sur le blog de Cloudflare, à cette adresse.

Les entrées suivantes désignent des types d'**attaques par injection** :

- **Injection de commandes** : le processus par lequel un acteur malveillant exécute des commandes arbitraires au sein du système d'exploitation de l'hôte via une application vulnérable.
- **Cross-Site Scripting (XSS) :** cette faille de sécurité permet à un acteur malveillant d'injecter des scripts au sein d'une application web, côté client, dans le but d'accéder directement à des informations importantes, d'usurper l'identité des utilisateurs ou de piéger ces derniers afin de les amener à révéler des informations importantes.
- **Injection SQL (SQLi) :** cette méthode permet aux acteurs malveillants d'exploiter les vulnérabilités inhérentes à la manière dont une base de données exécute les requêtes de recherche. Le SQLi permet ainsi d'accéder à des informations sans autorisation, de modifier ou de créer de nouvelles autorisations utilisateur, voire de manipuler ou détruire des données sensibles.

Requête HTTP : le moyen par lequel les plateformes de communication Internet, comme les navigateurs et les applications web, demandent les informations dont elles ont besoin pour charger une ressource.

Trafic atténué : toute requête HTTP* émanant d'un utilisateur à laquelle la plateforme Cloudflare a appliqué une action de « déconnexion ». Il peut s'agir d'actions de type `BLOCK` (Bloquer), `CHALLENGE` (Tester, par exemple, via des tests CAPTCHA ou basés sur le JavaScript). Le trafic atténué ne comprend pas les requêtes s'étant vu appliquer les actions suivantes : `LOG` (Journaliser), `SKIP` (Ignorer), `ALLOW` (Autoriser).

Contrôle du volume de requêtes : une technique utilisée par les systèmes informatiques pour contrôler la vitesse à laquelle les requêtes sont traitées. Elle peut être employée en tant que mesure de sécurité pour empêcher les attaques sur les API ou limiter l'utilisation de ressources au sein de vos serveurs d'origine.

Remote Code Execution (RCE, exécution de code distant) : le processus par lequel un acteur malveillant exécute du code malveillant sur les machines ou le réseau d'une entreprise. La capacité à exécuter du code contrôlé par le pirate peut être utilisée à diverses fins, notamment le déploiement de logiciels malveillants supplémentaires ou le vol de données sensibles.

Validation de schéma : si une requête d'API ne se conforme pas au schéma de l'API, cette dernière peut réagir de manière inattendue (en révélant des données confidentielles, par exemple). La validation de schéma permet à une API d'abandonner ces requêtes.

Vulnérabilités zero-day : il s'agit des vulnérabilités inconnues des concepteurs de l'application et qui ne disposent donc pas d'un correctif. Les acteurs malveillants cherchent à exploiter ces vulnérabilités aussi vite que possible.

Descriptions des codes de statut HTTP

Les exemples de codes de statut ci-dessous (qui désignent les erreurs les plus courantes concernant les API, comme expliqué dans la section 8) détaillent la manière dont Cloudflare interprète le protocole de suivi des normes Internet pour les codes de réponse HTTP. Veuillez vous référer à l'édition en cours des « normes de protocole Internet officielles » (STD 1) pour connaître l'état de normalisation et le statut de ce protocole.

429 signifie **Too Many Requests** (Trop de requêtes). Le client a envoyé un trop grand nombre de requêtes pendant la période spécifiée par le serveur. Ce code est également connu sous le nom de « contrôle du volume de requêtes ». Le serveur peut répondre par des informations permettant à l'entité effectuant la requête de réessayer après une période spécifiée.

400 signifie **Bad Request** (Requête incorrecte). Le client a envoyé une requête incorrecte au serveur. Il s'agit d'une erreur du client : malformation au niveau de la syntaxe de la requête, requête invalide, structure invalide du message ou routage trompeur d'une requête.

404 signifie **Not Found** (Ressource non trouvée). Le serveur d'origine n'a pas réussi ou n'a pas souhaité trouver la ressource demandée. Cette erreur implique généralement que le serveur hôte n'a pas reconnu l'URL de l'API, un problème pour lequel il existe plusieurs causes possibles.

401 signifie **Unauthorized** (Utilisateur non authentifié). Les identifiants de l'utilisateur n'existent pas ou ils ne contiennent pas le niveau d'accès approprié pour la ressource demandée.

403 signifie **Forbidden** (Accès interdit). Cloudflare renverra une réponse 403 si la requête a enfreint soit une règle WAF gérée activée par défaut pour l'ensemble des domaines en cloud orange Cloudflare soit une règle WAF gérée activée pour cette zone particulière. Si vous vous retrouvez en présence d'une erreur 403 dépourvue de la marque Cloudflare, cette dernière aura toujours été renvoyée directement par le serveur web d'origine, pas par Cloudflare. De même, elle sera généralement liée aux règles d'autorisation en vigueur sur votre serveur.

500 signifie **Internal Server Error** (Erreur interne du serveur). Il s'agit d'un message générique renvoyé en cas d'erreurs inattendues côté serveur.

422 signifie **Unprocessable Content** (Impossible de traiter le contenu). La requête contenait des erreurs d'ordre sémantique.

503 signifie **Service Unavailable** (Service indisponible). Le serveur pourrait être hors ligne en raison d'une opération de maintenance ou votre serveur web d'origine est surchargé.

430 signifie **Request Header Fields Too Large** (Les champs d'en-tête de la requête dépassent la taille maximale). Ce code d'erreur non officiel est utilisé par Shopify pour indiquer que la requête n'a pas été acceptée, car elle pourrait être malveillante. Shopify a répondu en la rejetant afin de protéger l'application d'attaques possibles.

402 signifie **Payment Required** (Paiement nécessaire). Ce code n'est pas largement utilisé, mais certaines plateformes s'en servent en cas de dépassement de limites quotidiennes ou de problèmes concernant un paiement.

Notes de fin

1. Le réseau Cloudflare diffuse en moyenne 50 millions de requêtes HTTP par seconde, avec plus de 70 millions de requêtes par seconde en période de pointe. Entre le 1er octobre 2022 et le 31 août 2023, le trafic d'API assorti de réponses fructueuses (code de statut 200) représentait entre 53,1 % et 60,1 % du trafic HTTP dynamique de Cloudflare. Le contenu dynamique désigne le contenu qui change en fonction de facteurs spécifiques à l'utilisateur, comme l'heure de la visite, son emplacement géographique et son appareil.
2. Pour les points de terminaison d'API REST, la fonctionnalité d'identification d'API de Cloudflare a découvert en moyenne, grâce au Machine Learning, 30,7 % de points de terminaison supplémentaires (260 contre 199) que ce que nous avions identifié à l'aide des identifiants de session fournis par les clients sur l'ensemble des domaines/zones des clients, et ce par compte.
3. Chiffres basés sur les codes de statut HTTP hors 2xx les plus courants (incluant donc les erreurs 4xx et 5xx), en pourcentage de l'ensemble des erreurs HTTP concernant les API (statut du cache dynamique) entre le 1er octobre 2022 et le 31 août 2023.
4. Pour calculer le trafic d'API atténué, Cloudflare a pris en compte le pourcentage quotidien de trafic lié aux API atténué par un produit Cloudflare, ainsi que le pourcentage quotidien de trafic atténué par des règles gérées dans la catégorie des règles de pare-feu d'applications web (WAF).
5. Secteurs principaux (conformes à la catégorie sectorielle Salesforce de l'entreprise) dans lesquels le trafic lié aux API totalisait plus de 70 % de l'ensemble du trafic HTTP dynamique du secteur.
6. Chiffres basés sur la part de trafic d'API en Amérique du Nord, en Europe, en Amérique latine, en Océanie, en Asie, en Afrique et au Moyen-Orient ayant renvoyé des codes de réponse de réussite (200) pour l'ensemble du trafic HTTP dynamique traité par le réseau Cloudflare.
7. Cloudflare utilise deux méthodes d'identification des API : l'examen du trafic contenant un identifiant de session et l'utilisation de son moteur d'identification basé sur l'apprentissage automatique (Machine Learning), qui ne nécessite pas la présence d'un identifiant de session. 15 431 comptes présentaient des points de terminaison qui ont uniquement pu être identifiés grâce au Machine Learning.
8. Chiffres basés sur le nombre agrégé d'API par compte, ventilés par points de terminaison disposant d'un accès en écriture (PUT, POST, PATCH, DELETE), par rapport à ceux qui ne disposaient que d'un accès « lecture d'informations » (GET). Aux fins de ce rapport, Cloudflare a calculé le pourcentage de comptes au sein desquels les API GET totalisaient au moins 50 % du nombre total d'API de chaque client.
9. Chiffres basés sur le trafic d'API atténué pour le compte des clients par la catégorie de règles gérées du pare-feu WAF Cloudflare.
10. Chiffres basés sur le pourcentage quotidien moyen d'API calculé à partir du nombre de requêtes d'API au sein de la région/du pays du client ayant renvoyé des codes de réponse 200 et une réponse de cache dynamique (sur l'ensemble du trafic des codes de réponses 200 comportant un cache dynamique).
11. Chiffres basés sur la variation quotidienne du pourcentage de trafic lié aux API par rapport à la base de référence (moyenne) du trafic d'API quotidien à travers le monde.
12. Chiffres basés sur le trafic HTTP dynamique total du secteur par rapport aux autres secteurs (le « secteur » étant défini par la catégorie sectorielle Salesforce du compte du client).

Téléphone : +33 7 57 90 52 73
E-mail : enterprise@cloudflare.com

Site : www.cloudflare.com

REV : BDES-5122.2024JAN04