



Try Hack Me

BURP SUITE

Teste d'intrusions informatique



08 août 2022

By PLarox

Sommaire

1. Burp Suite : les Bases

1.1 Introduction	-----	p 2
1.2 Qu'est-ce que Burp Suite ?	-----	p 2
1.3 des fonctionnalités de la communauté Burp	-----	p 3
1.4 l'installation	-----	p 4
1.5 Le tableau de bord	-----	p 5
1.6 la navigation	-----	p 8
1.7 Options	-----	p 8
1.8 Présentation du proxy Burp	-----	p10
1.9 via le proxy (FoxyProxy)	-----	p11
1.10 HTTPS	-----	p16
1.11 Le navigateur Burp Suite	-----	p18
1.12 Étendue et ciblage	-----	p19
1.13 Plan du site	-----	p21

2. Burp Suite : Répéteur

2.1 Qu'est-ce que le répéteur ?	-----	p22
2.2 Utilisation de base	-----	p23
2.3 Views	-----	p25
2.4 Inspector	-----	p26

3. Burp Suite : Intruder

3.1 Qu'est-ce qu'Intruder ?	-----	p28
3.2 Positions	-----	p29
3.3 Sniper	-----	p30
3.4 Béliet	-----	p31
3.5 Fourchette	-----	p32
3.6 Bombe	-----	p33
3.7 Intruder Payload	-----	p34

4. Burp Suite : Autres modules

4.1 Présentation Décodeur	-----	p37
4.2 Encodage / Décodage	-----	p38
4.3 Hachage	-----	p41
4.4 Présentation Comparer	-----	p43
4.5 Présentation Séquencer	-----	p45
4.6 Séquencer Capture en direct	-----	p46
4.7 Séquencer Analyse	-----	p47

5. Burp Suite : Extender

5.1 L'interface de l'extendeur	-----	p49
5.2 Le BApp Store	-----	p50
5.3 Jython	-----	p51
5.4 L'API Burp Suite	-----	p52

1. Burp Suite : les bases

1.1 Introduction

Ce cours couvrira les bases de l'utilisation du cadre d'application Web Burp Suite.

Plus précisément, nous examinerons :

- Qu'est-ce que Burp Suite
- Un aperçu des outils disponibles dans le framework
- Installer Burp Suite pour vous-même
- Naviguer et configurer Burp Suite.

Nous présenterons également le cœur du framework Burp Suite : le Proxy de Burp. Cette partie est principalement conçue pour fournir une connaissance de base de Burp Suite qui peut ensuite être approfondie dans les autres Parties du module Burp ; en tant que tel, il sera beaucoup plus lourd en théorie que les Parties suivantes, qui adoptent une approche plus pratique. Il est conseillé de lire les informations ici et suivez vous-même avec une copie de l'outil si vous n'avez jamais utilisé Burp Suite auparavant. L'expérimentation est essentielle : utilisez ces informations en même temps que vous jouez autour avec l'application pour vous-même pour construire une base en utilisant le cadre, qui peut ensuite être construit dans des pièces ultérieures.

1.2 Qu'est-ce que Burp Suite ?

En termes simples : Burp Suite est un framework écrit en Java qui vise à fournir un guichet unique pour les tests de pénétration des applications Web. À bien des égards, cet objectif est atteint car Burp est l'outil standard de l'industrie pour les évaluations pratiques de la sécurité des applications Web. Il est également très couramment utilisé lors de l'évaluation des applications mobiles, car les mêmes fonctionnalités qui la rendent si attrayante pour les tests d'applications Web se traduisent presque parfaitement par le test des API (Suite programmation **d'interfaces** des applications) qui alimentent la plupart des applications mobiles.

Au niveau le plus simple, Burp peut capturer et manipuler tout le trafic entre un attaquant et un serveur Web : c'est le cœur du framework. Après avoir capturé les demandes, nous pouvons choisir de les envoyer à diverses autres parties du cadre Burp Suite - nous couvrirons certains de ces outils dans les salles à venir. Cette capacité à intercepter, afficher et modifier les requêtes Web avant qu'elles ne soient envoyées au serveur cible (ou, dans certains cas, les réponses avant qu'elles ne soient reçues par notre navigateur), rend Burp Suite parfait pour tout type de test manuel d'applications Web.

Il existe différentes éditions de Burp Suite disponibles. Nous travaillerons avec **IBurp Suite Community**, car elle est gratuite pour toute utilisation (légale) non commerciale. Les éditions Burp Suite *Professional* et *Enterprise* nécessitent toutes deux des licences coûteuses, mais sont dotées de fonctionnalités supplémentaires puissantes :

- **Burp Suite Professional** est une version sans restriction de Burp Suite Community. Il est livré avec des fonctionnalités telles que :
 - Un scanner de vulnérabilité automatisé
 - Un fuzzer/bruteforcer qui n'est pas limité en débit
 - Enregistrer des projets pour une utilisation future ; génération de rapports
 - Une API intégrée pour permettre l'intégration avec d'autres outils
 - Accès illimité pour ajouter de nouvelles extensions pour une plus grande fonctionnalité
 - Accès au collaborateur Burp Suite (fournissant effectivement un attrape-requêtes unique auto-hébergé ou exécuté sur un serveur appartenant à Portswigger)

En bref, Burp Pro est un *extrêmement* puissant - c'est pourquoi il est proposé à un prix de 319 £ / 399 \$ par utilisateur pour un abonnement d'un an. Pour cette raison, Burp Pro n'est généralement utilisé que par des professionnels (les licences étant souvent fournies par les employeurs).

- **Burp Suite Enterprise** est légèrement différent. Contrairement aux éditions communautaires et professionnelles, Burp Enterprise est utilisé pour l'analyse continue. Il fournit un scanner automatisé qui peut analyser périodiquement les applications Web à la recherche de vulnérabilités de la même manière qu'un logiciel comme [Nessus](#) effectue une analyse automatisée de l'infrastructure. Contrairement aux autres éditions de Burp Suite qui vous permettent d'effectuer des attaques manuelles à partir de votre propre ordinateur, Enterprise se trouve sur un serveur et analyse en permanence les applications Web cibles à la recherche de vulnérabilités.

En raison des coûts prohibitifs liés à l'une ou l'autre de ces éditions de Burp Suite, nous nous en tiendrons à l'ensemble de fonctionnalités de base fourni par Burp Suite Community.

1.3 Des fonctionnalités de la communauté Burp

Bien que Burp Community dispose d'un ensemble de fonctionnalités relativement limité par rapport à l'édition professionnelle, il dispose toujours de nombreux outils superbes. Ceux-ci inclus :

- **Proxy** : L'aspect le plus connu de Burp Suite, le Burp Proxy nous permet d'intercepter et de modifier les requêtes/réponses lors de l'interaction avec les applications Web.
- **Répéteur** : la deuxième fonctionnalité Burp la plus connue - Répéteur - nous permet de capturer, de modifier, puis de renvoyer la même demande plusieurs fois. Devons créer une charge utile par essais et erreurs (par exemple, dans une injection SQLi - **S** tructured **Q** uery **L**anguage Injection **nous**) ou lors du test de la fonctionnalité d'un point de terminaison pour les défauts.
- **Intruder** : bien que très limité dans Burp Community, Intruder nous permet de pulvériser un point de terminaison avec des requêtes. Ceci est souvent utilisé pour les attaques par force brute ou pour fuzzer les terminaux.
- **Décodeur** : Bien que moins utilisé que les fonctionnalités mentionnées précédemment, Decoder fournit toujours un service précieux lors de la transformation des données, soit en termes de décodage des informations capturées, soit de codage d'une charge utile avant de l'envoyer à la cible. Bien qu'il existe d'autres services disponibles pour faire le même travail, le faire directement dans Burp Suite peut être très efficace.
- **Comparer** : Comme son nom l'indique, Comparer nous permet de comparer deux données au niveau du mot ou de l'octet. Encore une fois, ce n'est pas quelque chose qui est unique à Burp Suite, mais être capable d'envoyer des données (potentiellement très volumineuses) directement dans un outil de comparaison avec un seul raccourci clavier peut accélérer considérablement les choses.

- **Séquenceur** : nous utilisons généralement Séquencer pour évaluer le caractère aléatoire des jetons tels que les valeurs des cookies de session ou d'autres données générées soi-disant aléatoires. Si l'algorithme ne génère pas de valeurs aléatoires sécurisées, cela pourrait ouvrir des voies d'attaque dévastatrices.

En plus de la myriade de fonctionnalités intégrées, la base de code Java rend très facile l'écriture d'extensions à ajouter aux fonctionnalités du cadre Burp. Ceux-ci peuvent être écrits en Java, Python (en utilisant l'interpréteur Java Jython) ou Ruby (en utilisant l'interpréteur Java JRuby). Burp Suite extension module peut rapidement et facilement charger des extensions dans le framework, ainsi que fournir une place de marché pour télécharger des modules tiers (appelés le "BApp Store"). Bien que bon nombre de ces extensions nécessitent une licence professionnelle pour être téléchargées et ajoutées, il en existe encore un bon nombre qui peuvent être intégrées à Burp Community. Par exemple, nous pouvons souhaiter étendre la fonctionnalité de journalisation intégrée de Burp Suite avec le Logger++ module

1.4 Installation

Burp Suite est l'un de ces outils très utiles, que vous évaluiez explicitement une application Web ou mobile pour un pentest/bug bounty, ou que vous vouliez simplement déboguer une nouvelle fonctionnalité dans une application Web que vous développez. Pour cette raison, il est important de savoir comment installer Burp Suite sur une variété de plates-formes, plutôt que de simplement l'utiliser dans un système d'exploitation de pentesting tel que Kali ou Parrot. Vous ne savez jamais quand vous pourriez en avoir besoin !

Heureusement, PortSwigger a rendu l'installation de Burp Suite extrêmement facile sur Linux, macOS et Windows, en fournissant des installateurs dédiés pour les trois. En tant qu'application Java, Burp peut également être téléchargé en tant qu'archive JAR et s'exécuter efficacement sur tout ce qui prend en charge un environnement d'exécution Java.

Burp Suite est pré-emballé avec Kali Linux, vous ne devriez donc pas avoir besoin de l'installer là-bas. Si, pour une raison quelconque, Burp est absent de votre installation Kali, vous pouvez facilement l'installer à partir du Kali aptdépôts.

Pour les autres systèmes, nous pouvons télécharger des programmes d'installation à partir de la [page de téléchargement de Burp Suite](#).

Dans les menus déroulants, nous pouvons sélectionner notre système d'exploitation, ainsi que si nous voulons Burp Suite Community ou Burp Suite Professional :

Professional / Community 2021.8

Stable

05 August 2021 at 21:00 UTC



Burp Suite Community Edition Linux (64-bit)

Download

hide checksums

SHA256: f5ec72f7abcf53d55f39c0f7f87a9fc1cda6f27f3886268fe15553059be4f097
MD5: dab3b90335898090e6868230db4efb6

Nous pouvons ensuite cliquer sur le bouton "Télécharger" pour commencer à télécharger le programme d'installation de Burp Suite. Quel que soit le système d'exploitation que vous utilisez, **assurez-vous d'utiliser Burp Suite Community Edition.**

Une fois que nous avons vérifié l'intégrité de notre téléchargement, nous pouvons l'installer de la manière normale pour notre système d'exploitation (par exemple, exécuter l'exécutable sous Windows ou exécuter le script depuis le terminal avec `sudo` sous Linux).

Remarque : *Si vous installez sous Linux, vous pouvez choisir d'installer avec ou sans autorisations de super-utilisateur. Si vous décidez ne pas utiliser `sudo` lors de l'exécution du script, Burp Suite sera installé dans votre répertoire personnel à `~/BurpSuiteCommunity/BurpSuiteCommunity` et ne sera pas ajouté à votre PATH.*

L'assistant d'installation est très intuitif. Il est généralement prudent d'accepter les valeurs par défaut suggérées, quel que soit votre système d'exploitation ; cependant, il est toujours judicieux de lire attentivement le programme d'installation.

Avec Burp Suite installé, nous pouvons maintenant démarrer l'application. La première fois que nous l'utiliserons, Burp Suite nous demandera de lire et d'accepter ses termes et conditions ; assurez-vous de le faire avant de les accepter ou de les refuser !

Avec les termes et conditions acceptés, on nous présente un autre menu. Nous reviendrons sur cela dans la tâche suivante.

1.5 Le tableau de bord

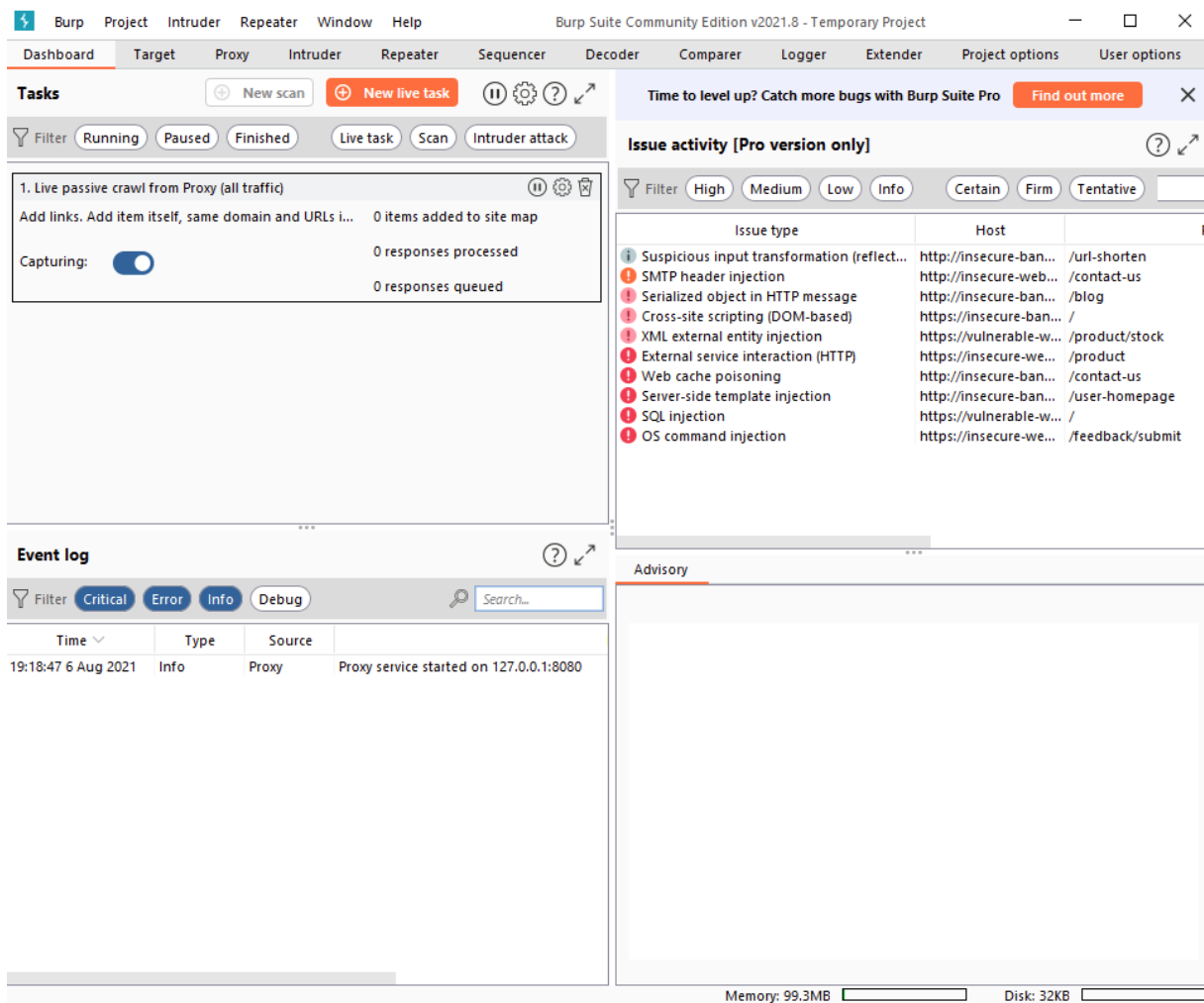
Lorsque nous ouvrons Burp Suite et avons accepté les termes et conditions, nous rencontrons une fenêtre nous demandant de sélectionner le type de projet.

Cette fenêtre ne nous donne pas beaucoup d'options dans Burp Community. Burp Pro nous permettrait d'enregistrer notre travail sur le disque ou de charger un projet précédemment enregistré à ce stade. Cependant, tout ce que nous pouvons faire ici est de cliquer sur "Suivant".

La fenêtre suivante nous permet de choisir une configuration pour Burp Suite. Laisser cela par défaut est parfait pour la plupart des situations :

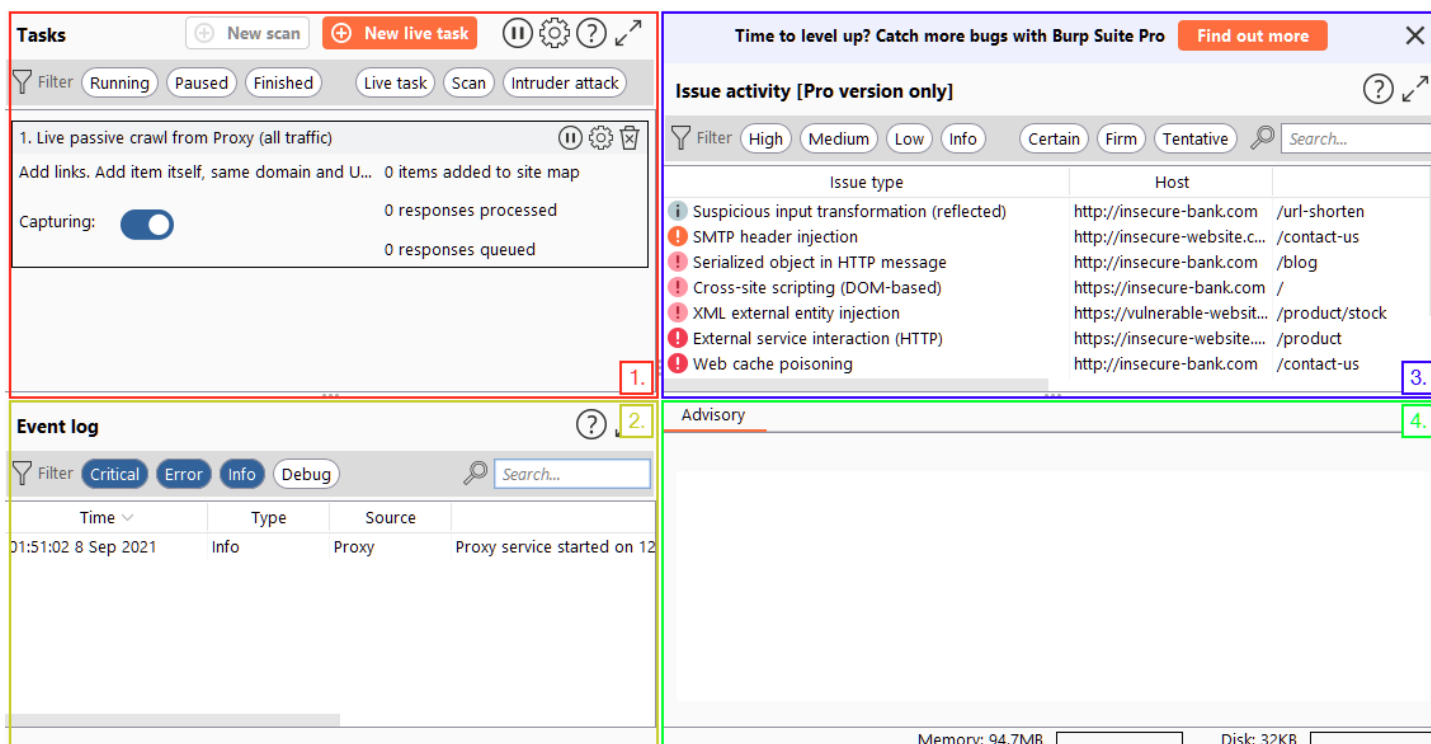
Cliquez sur "Démarrer Burp", et l'interface principale de Burp Suite s'ouvrira !

La première fois que vous ouvrez Burp Suite, un écran d'options d'entraînement peut s'afficher. Ceux-ci valent la peine d'être lus si vous avez le temps.



Si ce n'est pas le cas (et dans toutes les sessions ultérieures, quoi qu'il en soit), le tableau de bord Burp légèrement intimidant vous sera présenté :

Ne vous inquiétez pas si cela n'a pas encore trop de sens - cela le sera bientôt ! En bref, l'interface du tableau de bord est divisée en quatre quadrants :



1. Le menu Tâches nous permet de définir les tâches d'arrière-plan que Burp Suite exécutera pendant que nous utilisons l'application. La version Pro nous permettrait également de créer des scans à la demande. Le "Live Passive Crawl" par défaut (qui enregistre automatiquement les pages que nous visitons) sera plus que convenable pour nos utilisations dans ce module.
2. Le journal des événements nous indique ce que fait Burp Suite (par exemple, le démarrage du proxy), ainsi que des informations sur toutes les connexions que nous établissons via Burp.
3. La section Activité de problème est exclusive à Burp Pro. Cela ne nous donnera rien en utilisant Burp Community, mais dans Burp Professional, il listerait toutes les vulnérabilités trouvées par le scanner automatisé. Ceux-ci seraient classés par gravité et filtrables en fonction de la certitude de Burp que le composant est vulnérable.
4. La section Conseils donne plus d'informations sur les vulnérabilités trouvées, ainsi que des références et des suggestions de corrections. Ceux-ci pourraient ensuite être exportés dans un rapport.

Cliquer sur l'un des exemples de vulnérabilités dans la section Activité du problème nous donne une idée de ce à quoi cela ressemble :

Advisory Request Collaborator DNS interaction

! OS command injection

Issue: **OS command injection**
 Severity: **High**
 Confidence: **Certain**
 Host: **https://insecure-website.com**
 Path: **/feedback/submit**

Issue detail

The **subject** parameter appears to be vulnerable to OS command injection attacks. It is possible to use various shell metacharacters to inject arbitrary OS commands. The command output does not appear to

Dans les différents onglets et fenêtres de Burp Suite, vous trouverez de petites icônes d'aide : un point d'interrogation dans un cercle.

Cliquer dessus ouvrira une nouvelle fenêtre contenant de l'aide pour la section, par exemple :

Ceux-ci sont extrêmement utiles si jamais vous êtes bloqué et que vous ne savez pas ce que fait une fonctionnalité, alors faites-en bon usage !

burp-documentation.local/burp/documentation/desktop/dashboard/inde... — □ ×

Support Center >> Documentation >> Desktop editions >> Dashboard

Professional Community

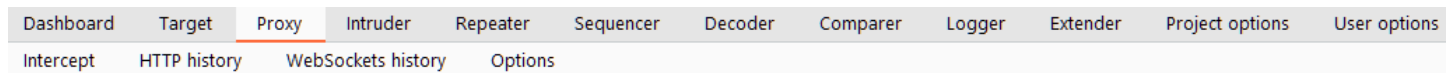
Dashboard

Burp's dashboard lets you control and monitor Burp's automated activity:

- You can **launch a scan** of a website by clicking the "New scan" button.

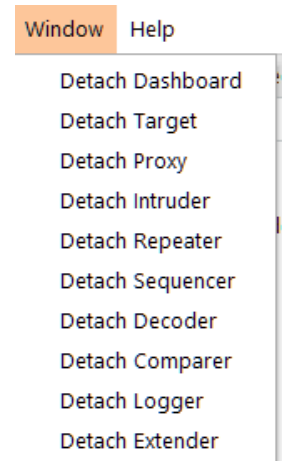
1.6 la navigation

La navigation dans l'interface graphique Burp Suite par défaut se fait entièrement à l'aide des barres de menus supérieures :



Ceux-ci vous permettent de basculer entre les modules (le long de la rangée supérieure de l'image jointe). Si le module sélectionné comporte plusieurs sous-onglets, ceux-ci peuvent être sélectionnés à l'aide d'une deuxième barre de menu qui apparaît directement sous la barre d'origine (la rangée inférieure de l'image ci-dessus). Il est courant que des paramètres spécifiques au module soient fournis dans ces sous-onglets (comme c'est le cas avec les options de proxy ci-dessus).

Les onglets peuvent également être affichés dans des fenêtres séparées si vous préférez afficher plusieurs onglets séparément. Cela peut être fait en cliquant sur "Fenêtre" dans le menu de l'application en haut de l'écran, puis en choisissant de "Détacher" les onglets :



Ceux-ci peuvent être remis en place de la même manière.

En plus de la barre de menus, Burp Suite dispose également de raccourcis clavier qui permettent une navigation rapide vers les onglets clés. Par défaut, ce sont :

Raccourci Fait

- Ctrl + Shift + D Passer au tableau de bord
- Ctrl + Shift + T Passer à l'onglet Cible
- Ctrl + Shift + P Passez à l'onglet Proxy
- Ctrl + Shift + I Basculer vers l'onglet Intrus
- Ctrl + Shift + R Passez à l'onglet Répéteur

1.7 Options

Avant de commencer à en savoir plus sur Burp Proxy, examinons les options disponibles pour configurer Burp Suite.

- Paramètres globaux peuvent être trouvés dans l'Options *utilisateur* le long de la barre de menu supérieure.
- Les paramètres spécifiques au projet se trouvent dans l'onglet Options *du projet*.

Les options fournies dans l'onglet Options utilisateur s'appliqueront à chaque fois que nous ouvrirons Burp Suite. En revanche, les options du projet ne s'appliqueront qu'au *cours*. Étant donné que nous ne pouvons pas enregistrer de projets dans Burp Community, cela signifie que nos options de projet seront réinitialisées à chaque fois que nous fermerons Burp.

Il y a beaucoup trop d'options pour toutes les couvrir, alors regardons simplement les catégories disponibles et soulignons quelques-uns des paramètres les plus importants. Nous allons commencer par l'onglet d'Options *utilisateur*.

[Sommaire](#)

Dashboard
Target
Proxy
Intruder
Repeater
Sequencer
Decoder
Comparer
Logger
Extender
Project options
User options

Connections
TLS
Display
Misc

?
Platform Authentication

These settings let you configure Burp to automatically carry out platform authentication to destination web servers.

Note: these settings can be overridden for individual projects within project options.

☒ Do platform authentication

Add	Enabled	Destination host ^	Type	Username	Domain	Domain hostname
Edit						
Remove						

☐ Prompt for credentials on platform authentication failure

?
Upstream Proxy Servers

The following rules determine whether Burp sends each outgoing request to a proxy server, or directly to the destination web server. The first rule that matches each destination host will be used. To send all traffic to a single proxy server, create a rule with * as the destination host.

Note: these settings can be overridden for individual projects within project options.

Add	Enabled	Destination host	Proxy host	Proxy port	Auth type	Username
Edit						
Remove						
Up						
Down						

?
SOCKS Proxy

These settings let you configure Burp to use a SOCKS proxy. This setting is applied at the TCP level, and all outbound requests will be sent via this proxy. If you have configured rules for upstream HTTP proxy servers, then requests to upstream proxies will be sent via the SOCKS proxy configured here.

Note: these settings can be overridden for individual projects within project options.

Les paramètres ici s'appliquent globalement (c'est-à-dire qu'ils contrôlent l'application Burp Suite - pas seulement le projet). Cela dit, beaucoup d'entre eux peuvent être écrasés dans les paramètres du projet.

Il existe quatre sous-sections principales de l'onglet Options utilisateur :

- Les options du **Connexions** nous permettent de contrôler la manière dont Burp établit des connexions avec les cibles. Par exemple, nous pouvons définir un proxy pour que Burp Suite se connecte ; ceci est très utile si nous voulons utiliser Burp Suite via un pivot réseau.
- Le **TLS** nous permet d'activer et de désactiver diverses options TLS (**Transport Layer Security**), ainsi que de nous donner un endroit pour télécharger des certificats clients si une application Web nous oblige à en utiliser un pour les connexions.
- Un ensemble d'options essentiel : **Display** nous permet de changer l'apparence de Burp Suite. Les options ici incluent des choses comme changer la police et l'échelle, ainsi que définir le thème pour le cadre (par exemple le mode sombre) et configurer diverses options à faire avec le moteur de rendu dans Repeater (plus à ce sujet plus tard !).
- Le **Divers** contient une grande variété de paramètres, y compris le tableau des raccourcis clavier (HotKeys), qui nous permet de visualiser et de modifier les raccourcis claviers utilisés par Burp Suite. Il serait conseillé de vous familiariser avec ces paramètres, car l'utilisation des raccourcis clavier peut accélérer considérablement votre flux de travail.

Avec ces options, nous pouvons personnaliser notre installation Burp en fonction de nos préférences individuelles.

Passons maintenant à l'examen des configurations spécifiques au projet disponibles dans l'onglet *Options du projet*.

Il y a cinq sous-onglets ici :

- **Connections** contient bon nombre des mêmes options que la section équivalente de l'onglet Options utilisateur : elles peuvent être utilisées pour remplacer les paramètres à l'échelle de l'application. Par exemple, il est possible de définir un proxy uniquement pour le projet, en remplaçant tous les paramètres de proxy que vous avez définis dans l'onglet Options utilisateur. Il existe cependant quelques différences entre ce sous-onglet et celui des options utilisateur. Par exemple, l'option "Résolution du nom d'hôte" (vous permettant de mapper des domaines à des adresses IP directement dans Burp Suite) peut être très pratique - tout comme les paramètres "Requêtes hors champ", qui nous permettent de déterminer si Burp enverra des demandes à tout ce que vous ne ciblez pas explicitement (plus à ce sujet plus tard !).
- Le **HTTP** définit la manière dont Burp gère divers aspects du protocole HTTP : par exemple, s'il suit les redirections ou comment gérer les codes de réponse inhabituels.
- **TLS** nous permet de remplacer les options TLS à l'échelle de l'application, ainsi que de nous montrer une liste de certificats de serveur public pour les sites que nous avons visités.
- **Sessions** nous offre des options pour gérer les sessions. Il nous permet de définir comment Burp obtient, enregistre et utilise les cookies de session qu'il reçoit des sites cibles. Cela nous permet également de définir des macros que nous pouvons utiliser pour automatiser des choses telles que la connexion à des applications Web (nous donnant une session authentifiée instantanée, en supposant que nous avons des informations d'identification valides).
- Il y a moins d'options dans le **Divers** que dans l'onglet équivalent de la section "Options utilisateur". De nombreuses options ici ne sont également disponibles que si vous avez accès à Burp Pro (comme celles qui configurent Collaborator). Cela dit, il existe quelques options liées à la journalisation et au navigateur intégré (que nous examinerons dans quelques tâches) qui valent la peine d'être lues.

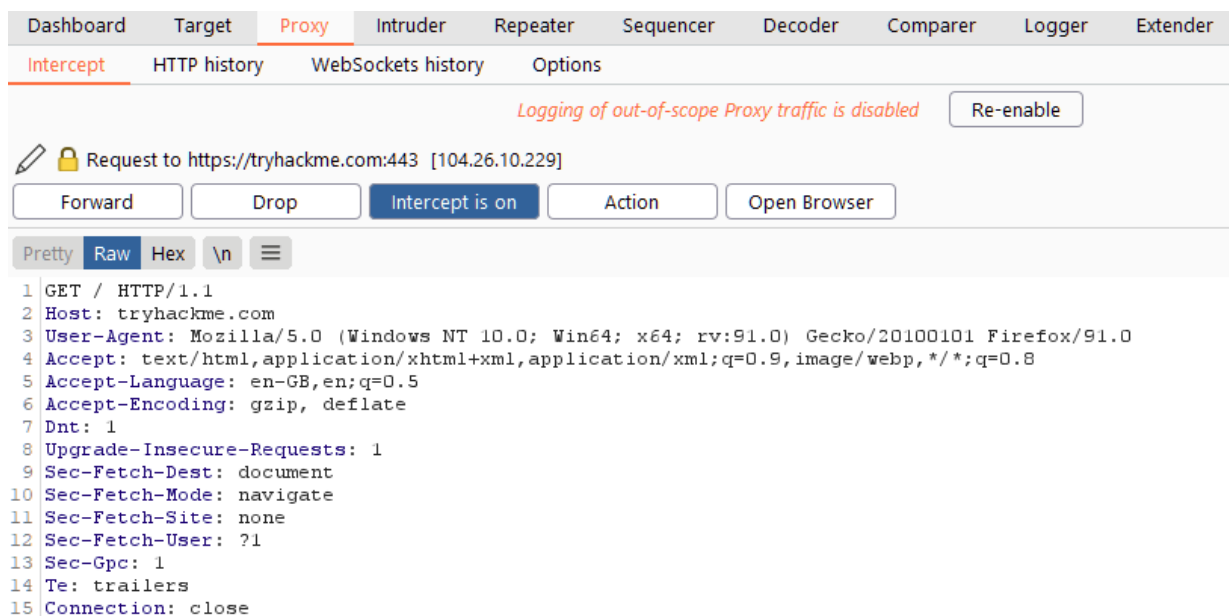
1.8 Présentation du proxy Burp

Le Burp Proxy est le plus fondamental (et le plus important !) des outils disponibles dans Burp Suite. Il nous permet de capter les demandes et les réponses entre nous et notre cible. Ceux-ci peuvent ensuite être manipulés ou envoyés à d'autres outils pour un traitement ultérieur avant d'être autorisés à continuer vers leur destination.

Par exemple, si nous faisons une demande à <https://tryhackme.com> via le proxy Burp, notre demande sera capturée et ne sera pas autorisée à continuer vers les serveurs TryHackMe jusqu'à ce que nous l'autorisions explicitement. Nous pouvons choisir de faire de même avec la réponse du serveur, bien que celle-ci ne soit pas active par défaut. Cette capacité à intercepter les requêtes signifie en fin de compte que nous pouvons prendre le contrôle total de notre trafic Web - une capacité inestimable lorsqu'il s'agit de tester des applications Web.

Il y a quelques configurations que nous devons faire avant de pouvoir utiliser le proxy, mais commençons par regarder l'interface.

Lorsque nous ouvrons l'onglet Proxy pour la première fois, Burp nous donne un tas d'informations utiles et de lecture de fond. Ces informations valent la peine d'être lues ; cependant, la vraie magie se produit après la capture d'une demande :



Avec le proxy actif, une demande a été faite au site Web TryHackMe. À ce stade, le navigateur à l'origine de la demande se bloquera et la demande apparaîtra dans l'onglet Proxy, nous donnant la vue illustrée dans la capture d'écran ci-dessus. Nous pouvons alors choisir de transférer ou de supprimer la demande (éventuellement après l'avoir modifiée). Nous pouvons également faire diverses autres choses ici, comme envoyer la requête à l'un des autres modules Burp, la copier en tant que commande CURL, l'enregistrer dans un fichier, et bien d'autres.

Lorsque nous avons fini de travailler avec le proxy, nous pouvons cliquer sur le bouton "interception est activée" pour désactiver l'interception, ce qui permettra aux demandes de passer par le proxy sans être arrêtées.

Burp Suite enregistrera toujours (par défaut) les requêtes effectuées via le proxy lorsque l'interception est désactivée. Cela peut être très utile pour revenir en arrière et analyser les demandes antérieures, même si nous ne les avons pas saisies spécifiquement lorsqu'elles ont été faites.

Burp capturera et enregistrera également la communication WebSocket, ce qui, encore une fois, peut être extrêmement utile lors de l'analyse d'une application Web.

Les logs sont consultables en se rendant dans les sous-onglets "Historique HTTP" et "Historique WebSockets" :

Dashboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project options	User options
Intercept	HTTP history	WebSockets history	Options								
Filter: Hiding CSS, image and general binary content											
#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension		
1	https://tryhackme.com	GET	/			200	33019	HTML		TryHackMe	
3	https://tryhackme.com	GET	/assets/pace/pace.js			200	27736	script	js		
5	https://cdnjs.cloudflare.com	GET	/ajax/libs/cookieconsent2/3.0.3/cookie...			200	20998	script	js		

Il convient de noter que toutes les requêtes capturées ici peuvent être envoyées à d'autres outils du framework en cliquant dessus avec le bouton droit de la souris et en choisissant "Envoyer à...". Par exemple, nous pourrions prendre une HTTP qui a déjà été transmise à la cible et l'envoyer à [Repeater](#).

Enfin, il existe également des options spécifiques au proxy, que nous pouvons voir dans le sous-onglet "Options".

Ces options nous donnent *beaucoup* de contrôle sur le fonctionnement du proxy, c'est donc une excellente idée de vous familiariser avec celles-ci.

Par exemple, le proxy n'interceptera pas les réponses du serveur par défaut, sauf si nous le lui demandons explicitement sur une base par requête. Nous pouvons remplacer le paramètre par défaut en cochant la case "Intercepter les réponses en fonction des règles suivantes" et en choisissant une ou plusieurs règles. La règle " Or Request Was Intercepted" est bonne pour intercepter les réponses à toutes les requêtes qui ont été interceptées par le proxy :

?

Intercept Server Responses

⚙

Use these settings to control which responses are stalled for viewing and editing in the Intercept tab.

☒ Intercept responses based on the following rules:

Add

Edit

Remove

Up

Down

Enabled	Operator	Match type	Relationship	Condition
<input checked="" type="checkbox"/>		Content type header	Matches	text
<input type="checkbox"/>	Or	Request	Was modified	
<input checked="" type="checkbox"/>	Or	Request	Was intercepted	
<input type="checkbox"/>	And	Status code	Does not match	^304\$
<input type="checkbox"/>	And	URL	Is in target scope	

☒ Automatically update Content-Length header when the response is edited

" And URL Is in target scope" est une autre très bonne règle par défaut ; nous examinerons la portée plus tard dans cette partie.

Vous pouvez créer vos propres règles pour la plupart des options de proxy, c'est donc une section où regarder autour de vous et expérimenter vous sera très utile !

Une autre section particulièrement utile de ce sous-onglet est la section "Match and Replace" ; cela vous permet d'effectuer des regex sur les requêtes entrantes et sortantes. Par exemple, vous pouvez modifier automatiquement votre agent utilisateur pour émuler un navigateur Web différent dans les requêtes sortantes ou supprimer tous les cookies définis dans les requêtes entrantes. Encore une fois, vous êtes libre de faire vos propres règles ici.

1.9 via le proxy (FoxyProxy)

Vous avez vu la théorie ; il est maintenant temps de commencer à utiliser le proxy pour vous-même.

Il existe deux façons de proxy notre trafic via Burp Suite.

1. Nous pourrions utiliser le navigateur intégré (nous aborderons cela dans une tâche ultérieure).
2. Nous pouvons configurer notre navigateur Web local pour proxy notre trafic via Burp ; c'est plus courant et ce sera donc l'objet de cette tâche.

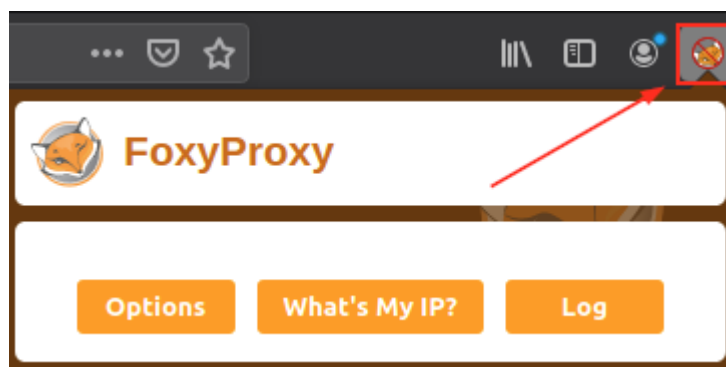
Le Burp Proxy fonctionne en ouvrant une interface web sur 127.0.0.1:8080 (par défaut). Comme l'implique le fait qu'il s'agit d'un "proxy", nous devons rediriger tout le trafic de notre navigateur via ce port avant de pouvoir commencer à l'intercepter avec Burp. Nous pouvons le faire en modifiant les paramètres de notre navigateur ou, plus communément, en utilisant une extension de navigateur Firefox appelée [FoxyProxy](#). FoxyProxy nous permet d'enregistrer des profils de proxy, ce qui signifie que nous pouvons basculer rapidement et facilement vers notre profil "Burp Suite" en quelques clics, puis désactiver le proxy tout aussi facilement.

Remarque : Toutes les instructions seront données avec Firefox à l'esprit, car il s'agit du navigateur par défaut pour Kali Linux.

Il existe deux versions de FoxyProxy : *Basic* et *Standard*. Les deux versions vous permettent de modifier vos paramètres de proxy à la volée ; cependant, FoxyProxy Standard vous donne beaucoup plus de contrôle sur le trafic envoyé via le proxy. Il vous permettra par exemple de définir des règles de pattern matching pour déterminer si une requête doit être proxy ou non : c'est plus compliqué que le simple proxy proposé par FoxyProxy basic.

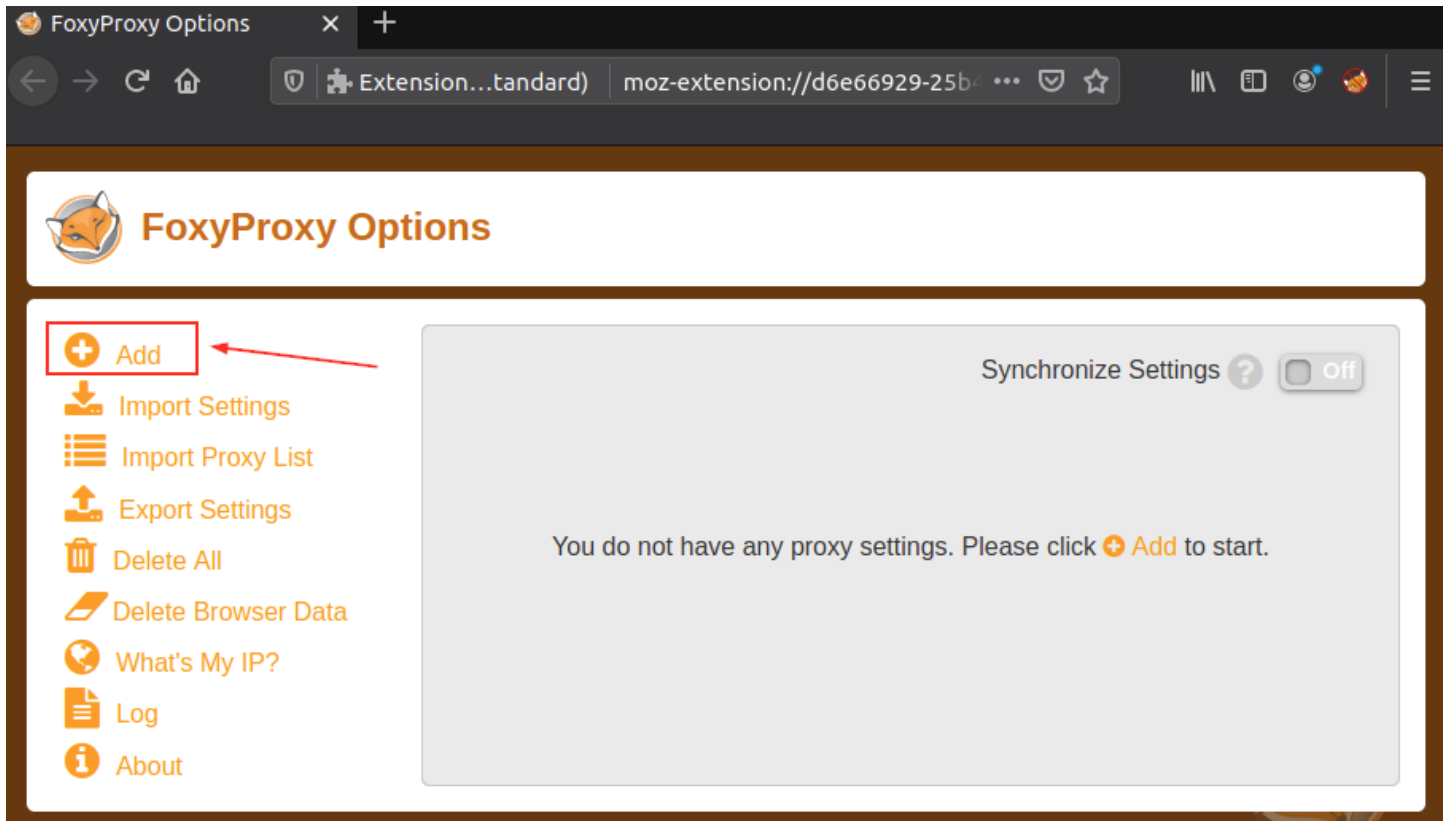
Si vous utilisez votre propre machine, vous pouvez télécharger FoxyProxy Basic [ici](#).

Une fois installé, un bouton devrait apparaître en haut à droite de l'écran qui vous permet d'accéder à vos configurations de proxy :



Il n'y a pas de configuration par défaut, alors cliquons sur le bouton "Options" pour créer notre configuration Burp Proxy.

Cela ouvrira un nouvel onglet de navigateur avec la page d'options de FoxyProxy :



Cliquez sur le bouton "Ajouter" et renseignez les valeurs suivantes :

- Titre : Burp (ou autre chose que vous préférez)
- Adresse IP : 127.0.0.1
- Port : 8080

Add Proxy

Title or Description (optional)
Burp

Color
#66cc66

Proxy Type
HTTP

Proxy IP address or DNS name ★
127.0.0.1

Port ★
8080

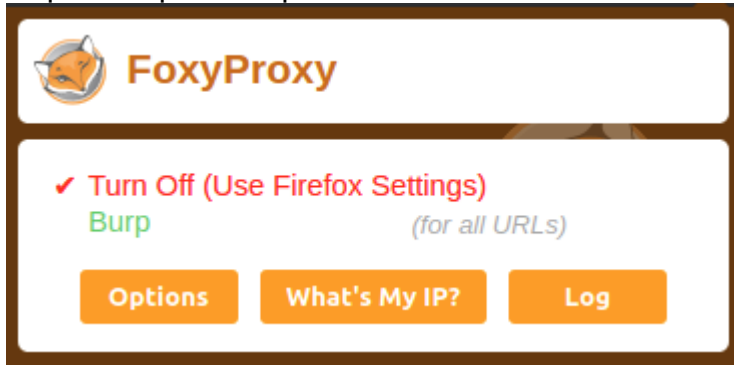
Username (optional)
username

Password (optional)

Cancel Save & Add Another Save

Cliquez maintenant sur "Enregistrer".

Lorsque vous cliquez sur l'icône FoxyProxy en haut de l'écran, vous verrez qu'il y a une configuration disponible pour Burp :

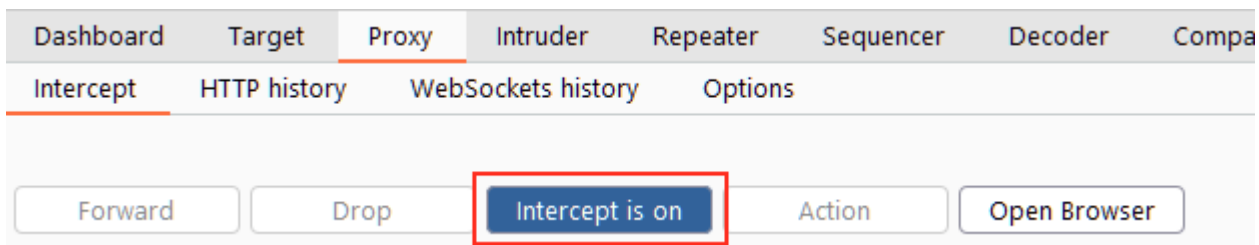


Si nous cliquons sur la configuration "Burp", notre navigateur commencera à diriger tout notre trafic via 127.0.0.1:8080. **Attention** : si Burp Suite n'est pas en cours d'exécution, votre navigateur ne pourra faire aucune requête lorsque cette configuration sera activée !

Activez cette configuration maintenant -- l'icône dans le menu devrait changer pour indiquer qu'un proxy est en cours d'exécution :



Ensuite, passez à Burp Suite et assurez-vous que l'interception est activée :



Maintenant, essayez d'accéder à la page d'accueil pour `http://MACHINE_IP/` dans Firefox. Votre navigateur devrait se bloquer et votre proxy se remplira avec les en-têtes de requête.

Félicitations, vous venez d'intercepter votre première requête !

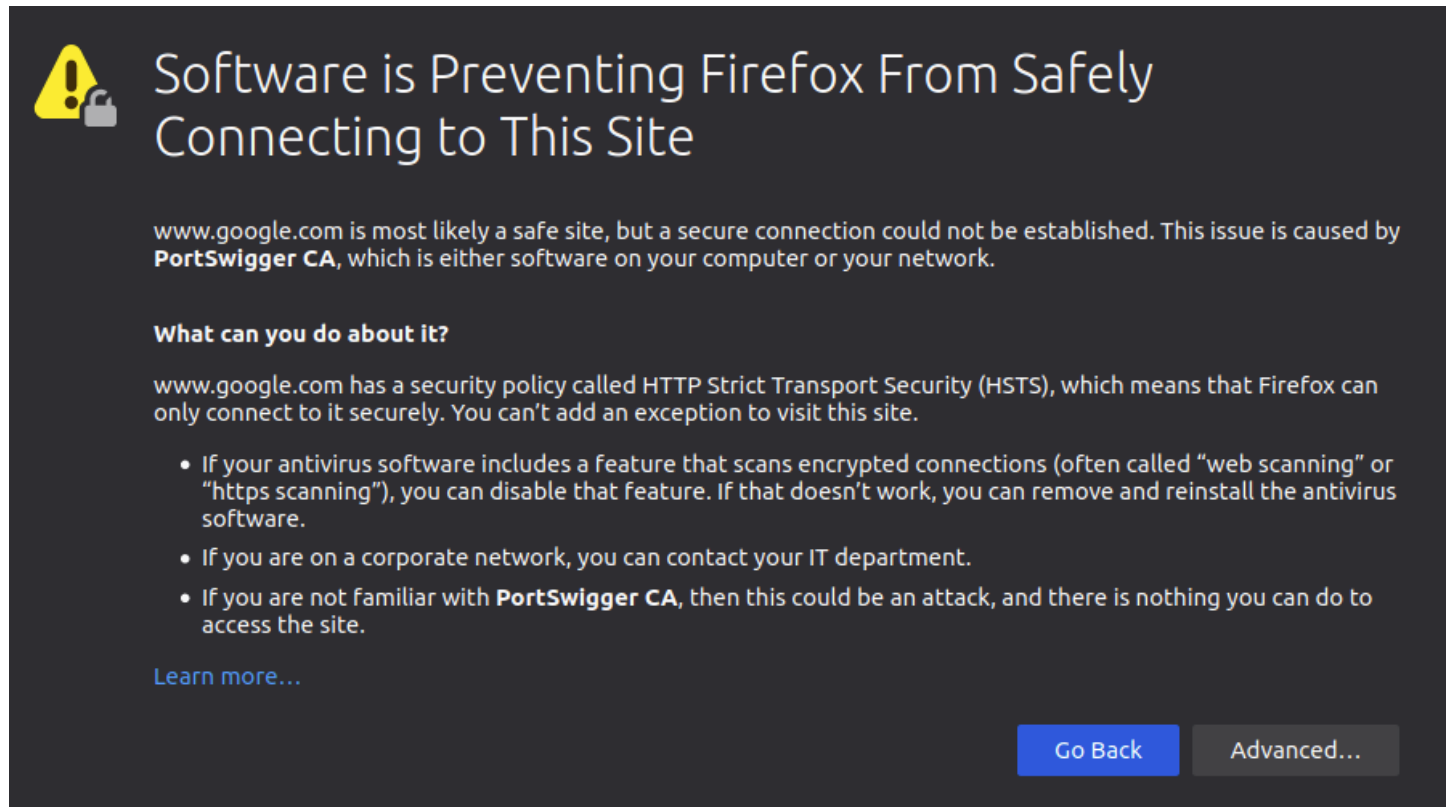
À partir de là, vous pouvez choisir de transférer ou de supprimer la demande. Vous pouvez également l'envoyer à un autre outil ou effectuer un certain nombre d'autres actions en cliquant avec le bouton droit sur la demande et en sélectionnant une option dans le menu contextuel.

N'oubliez pas : tant que vous êtes connecté au proxy et que l'interception de proxy est activée, votre navigateur se bloque chaque fois que vous faites une demande. Une erreur très courante lorsque vous apprenez à utiliser Burp Suite (et même plus tard !) Est de laisser accidentellement l'interception activée et donc de ne pas pouvoir faire de requêtes Web via votre navigateur. Si votre navigateur se bloque et que vous ne savez pas pourquoi : vérifiez votre proxy !

1.10 HTTPS

Génial, nous pouvons donc intercepter HTTP le trafic

Malheureusement, il y a un problème. Que se passe-t-il si nous naviguons vers un site avec TLS activé ? Par exemple, <https://google.com/>:



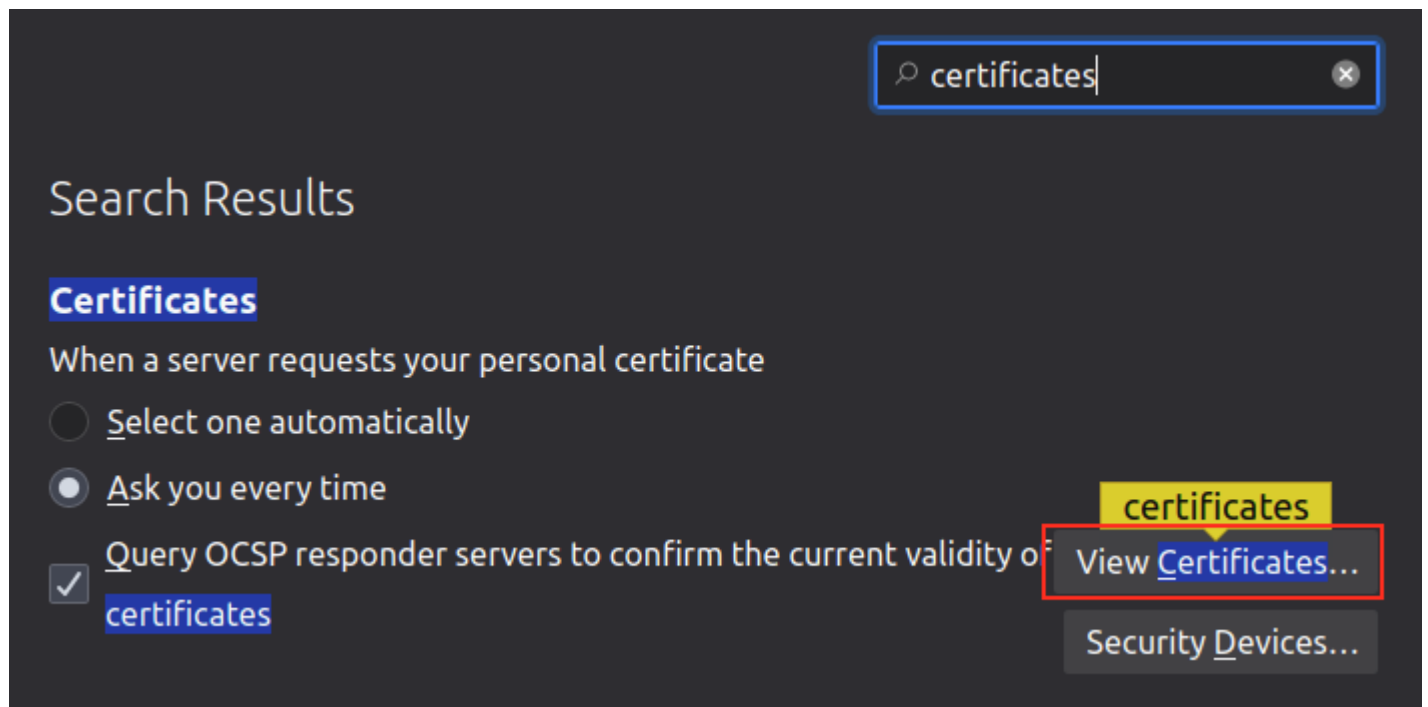
Nous obtenons une erreur.

Plus précisément, Firefox nous dit que l'autorité de **CA Portswigger certification**) n'est pas autorisée à sécuriser la connexion.

Heureusement, Burp nous propose un moyen simple de contourner ce problème. Nous devons faire en sorte que Firefox fasse confiance aux connexions sécurisées par les certificats Portswigger, nous allons donc ajouter manuellement le certificat CA à notre liste d'autorités de certification de confiance.

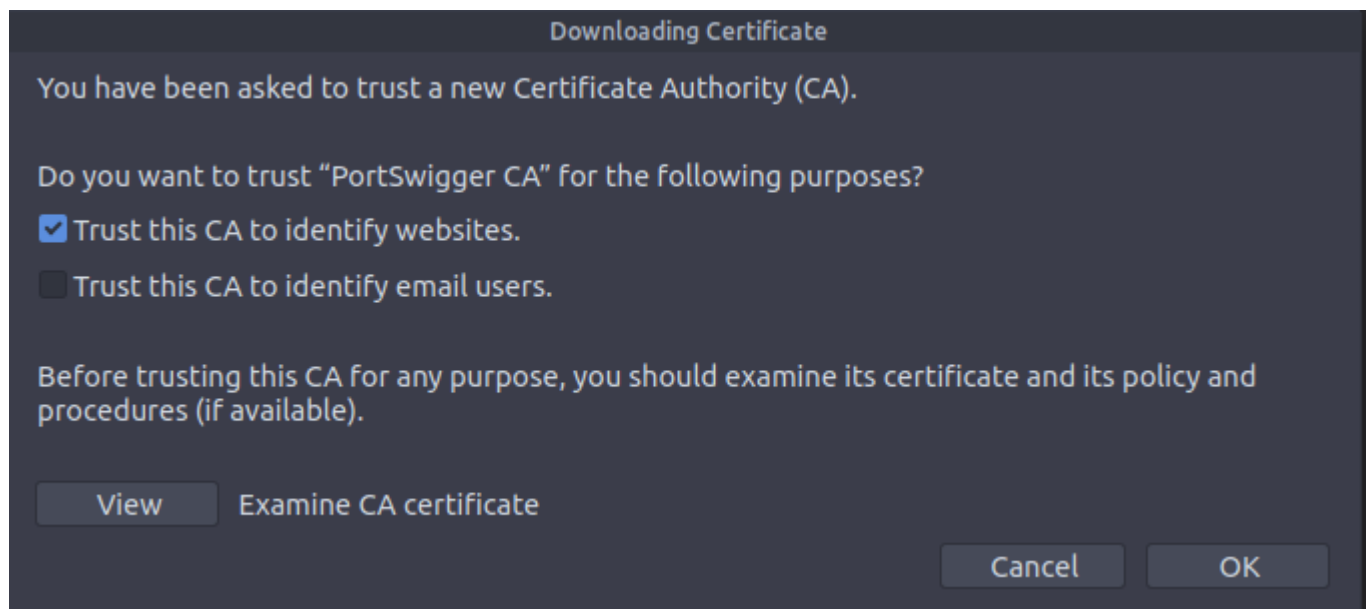
Tout d'abord, avec le proxy activé, dirigez-vous vers <http://burp/cert> ; cela téléchargera un fichier appelé **cacert.der**-- enregistrez-le quelque part sur votre machine.

Ensuite, tapez **about:preferences** dans votre barre de recherche Firefox et appuyez sur Entrée ; cela nous amène à la page des paramètres de FireFox. Recherchez la page "certificats" et nous trouvons l'option "Afficher les certificats":



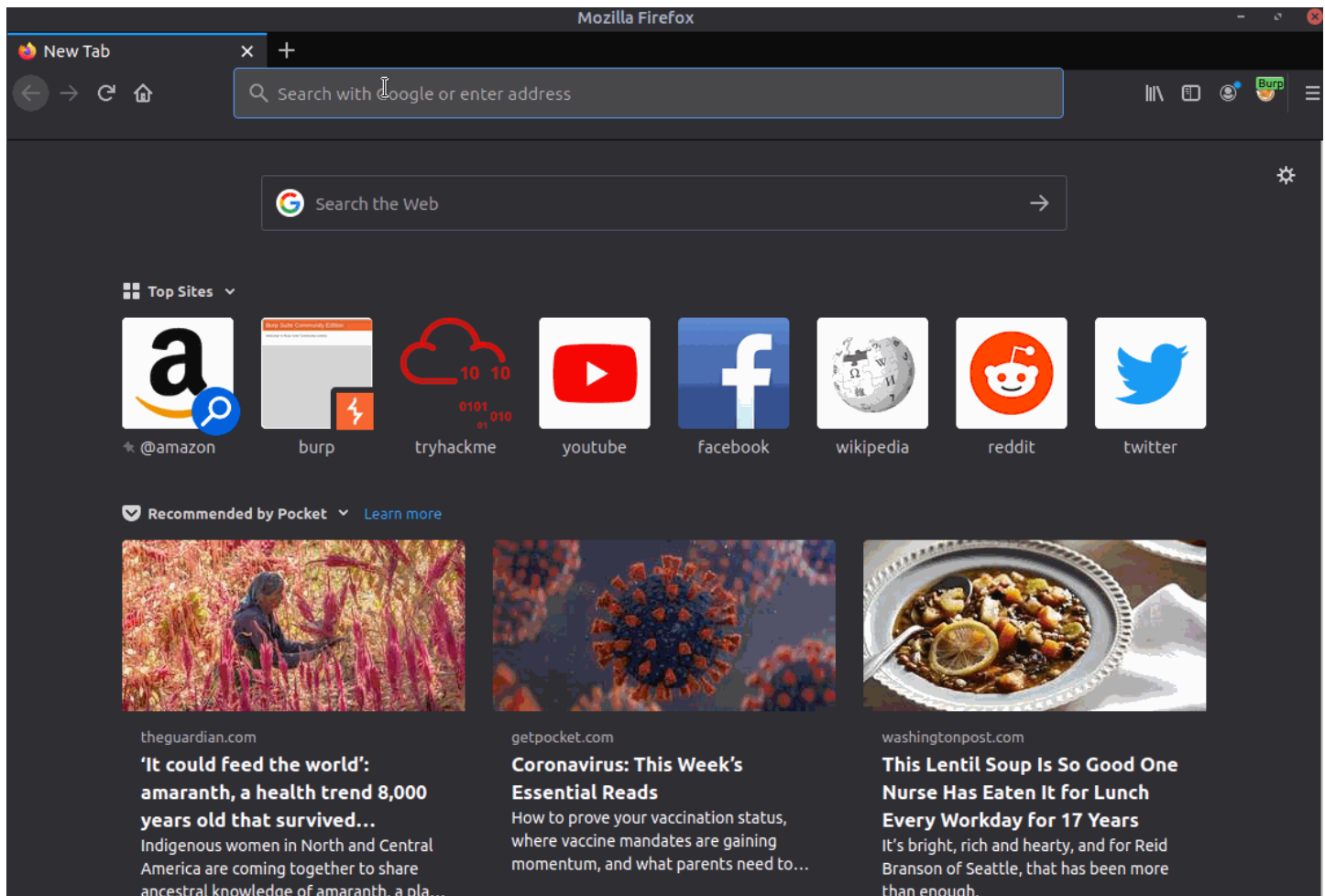
Cliquer sur le bouton "Afficher les certificats" nous permet de voir tous nos certificats CA de confiance. Nous pouvons enregistrer un nouveau certificat pour Portswigger en appuyant sur "Importer" et en sélectionnant le fichier que nous venons de télécharger.

Dans le menu qui s'affiche, sélectionnez "Faire confiance à cette autorité de certification pour identifier les sites Web", puis cliquez sur OK :



Nous devrions maintenant être libres de visiter tous les sites compatibles TLS que nous souhaitons !

La vidéo suivante montre le processus d'importation complet :



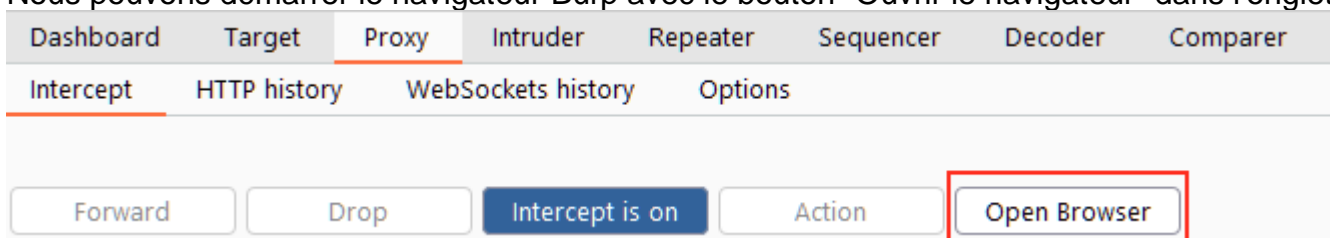
1.11 Le navigateur Burp Suite

Si les dernières tâches semblaient trop complexes, rassurez-vous, ce sujet sera beaucoup plus simple.

En plus de nous donner la possibilité de modifier notre navigateur Web habituel pour qu'il fonctionne avec le proxy, Burp Suite comprend également un navigateur Chromium intégré qui est préconfiguré pour utiliser le proxy sans aucune des modifications que nous devons faire.

Bien que cela puisse sembler idéal, il n'est pas aussi couramment utilisé que le processus détaillé dans les quelques tâches précédentes. Les gens ont tendance à s'en tenir à leur propre navigateur car cela leur donne beaucoup plus de personnalisation ; cependant, les deux sont des choix parfaitement valables.

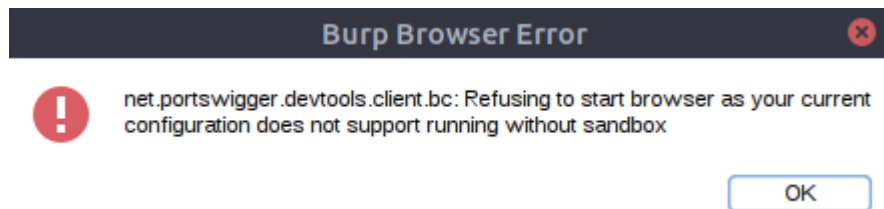
Nous pouvons démarrer le navigateur Burp avec le bouton "Ouvrir le navigateur" dans l'onglet proxy :



Une fenêtre Chromium va maintenant apparaître. Toutes les demandes que nous faisons dans ce domaine passeront par le proxy.

Remarque : Il existe de nombreux paramètres à régler avec le navigateur Burp dans les onglets *Options du projet* et *Options utilisateur*. Assurez-vous d'y aller les consulter !

Si nous fonctionnons sous Linux en tant qu'utilisateur, Burp Suite est incapable de créer un environnement sandbox pour démarrer le navigateur Burp, ce qui provoque une erreur et sa mort :



Il existe deux solutions simples à cela :

- **L'option intelligente :** nous pourrions créer un nouvel utilisateur et exécuter Burp Suite sous un compte à faibles privilèges.
- **L'option facile :** Nous pourrions aller à **Project options -> Misc -> Embedded Browser** et vérifier l'option **Allow the embedded browser to run without a sandbox**. Cocher cette option permettra au navigateur de démarrer, mais sachez qu'il est désactivé par défaut pour des raisons de sécurité : si nous sommes compromis en utilisant le navigateur, alors un attaquant aura accès à l'ensemble de notre machine.

1.12 Étendue et ciblage

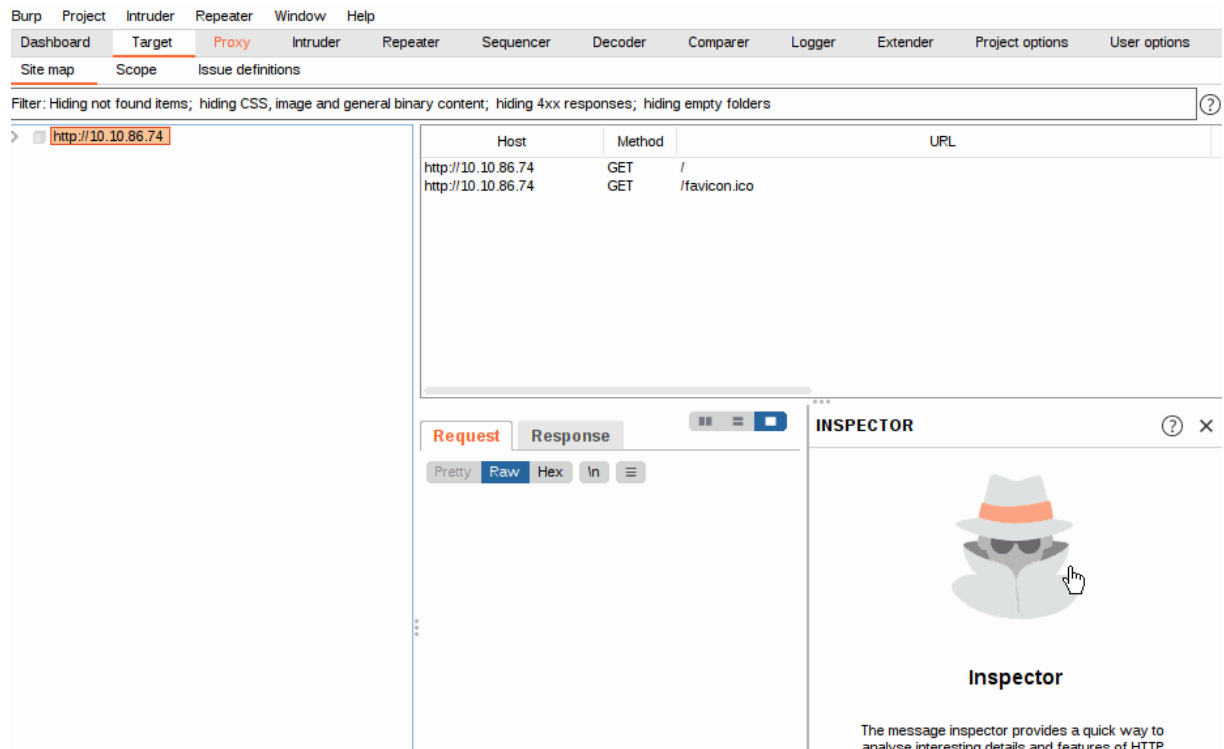
Enfin, nous arrivons à l'une des parties les plus importantes de l'utilisation du Burp Proxy : la portée.

Cela peut devenir extrêmement fastidieux d'avoir Burp capturant tout notre trafic. Lorsqu'il enregistre tout (y compris le trafic vers des sites que nous ne ciblons pas), il brouille les journaux que nous souhaiterons peut-être envoyer ultérieurement aux clients. En bref, permettre à Burp de *tout* peut rapidement devenir une énorme douleur.

Quelle est la solution ? Portée.

Définir une portée pour le projet nous permet de définir ce qui est proxy et enregistré. Nous pouvons restreindre Burp Suite à *seulement* cibler la ou les applications Web que nous voulons tester. La façon la plus simple de faire c'est en basculant vers l'onglet "Cible", en faisant un clic droit sur notre cible dans notre liste sur la gauche, puis en choisissant "Ajouter à la portée". Burp demandera alors nous indiquer si nous voulons arrêter de consigner tout ce qui n'est pas dans le champ d'application

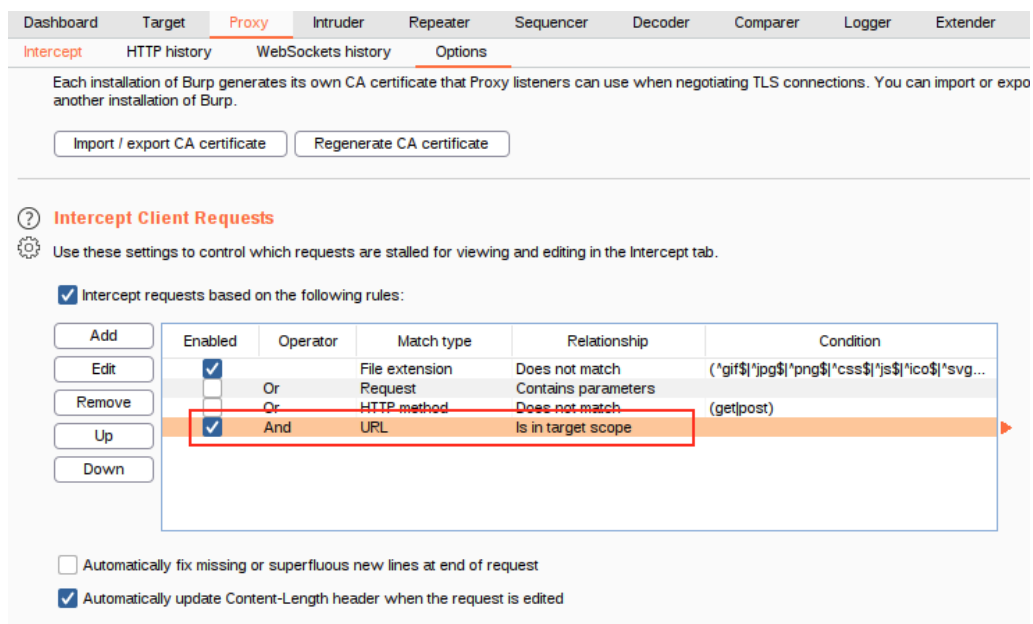
La plupart du temps, nous voulons choisir "oui" ici.



Nous pouvons maintenant vérifier notre portée en passant au sous-onglet "Scope" (comme indiqué dans le GIF ci-dessus).

Le sous-onglet Portée nous permet de contrôler ce que nous ciblons en *incluant* ou *excluant* domaines/IP. C'est une section très puissante, il vaut donc la peine de prendre le temps de s'habituer à l'utiliser.

Nous avons simplement choisi de désactiver la *journalisation* pour le trafic hors de portée, mais le proxy interceptera toujours tout. Pour désactiver cela, nous devons aller dans le sous-onglet Options de proxy et sélectionner "And URL Is in target scope" de la section Intercept Client Requests :



Avec Cette option sélectionnée, le proxy ignorera complètement tout ce qui n'est pas dans la portée, nettoyant considérablement le trafic passant par Burp.

[Sommaire](#)

1.13 Plan du site

Le contrôle de la portée peut être l'aspect le plus utile de l'onglet Cible, mais ce n'est en aucun cas la *seule* utilisation de cette section de Burp.

Il y a trois sous-onglets sous *Target* :

- **Plan du site** nous permet de cartographier les applications que nous ciblons dans une arborescence. Chaque page que nous visitons s'affichera ici, ce qui nous permettra de générer automatiquement un plan du site pour la cible simplement en parcourant l'application Web. Burp Pro nous permettrait également d'attraper les cibles automatiquement (c'est-à-dire de parcourir chaque page à la recherche de liens et de les utiliser pour cartographier la plus grande partie du site accessible au public en utilisant les liens entre les pages) ; cependant, avec Burp Community, nous pouvons toujours l'utiliser pour accumuler des données pendant que nous effectuons nos premières étapes d'énumération.
Le plan du site peut être particulièrement utile si nous voulons cartographier une API, car chaque fois que nous visitons une page, tous les points de terminaison de l'API dont la page récupère les données lors du chargement s'afficheront ici.
- **Portée** : Nous avons déjà vu le sous-onglet Portée -- il nous permet de contrôler la portée cible de Burp pour le projet.
- **Définitions des problèmes** : bien que nous n'ayons pas accès au scanner de vulnérabilité Burp Suite dans la communauté Burp, nous avons toujours accès à une liste de toutes les vulnérabilités qu'il recherche. La section Définitions des problèmes nous donne une énorme liste de vulnérabilités Web (avec des descriptions et des références) à partir de laquelle nous pouvons puiser si nous avons besoin de citations pour un rapport ou d'aide décrivant une vulnérabilité.

2.Burp Suite : Répéteur

2.1 Qu'est-ce que le répéteur ?

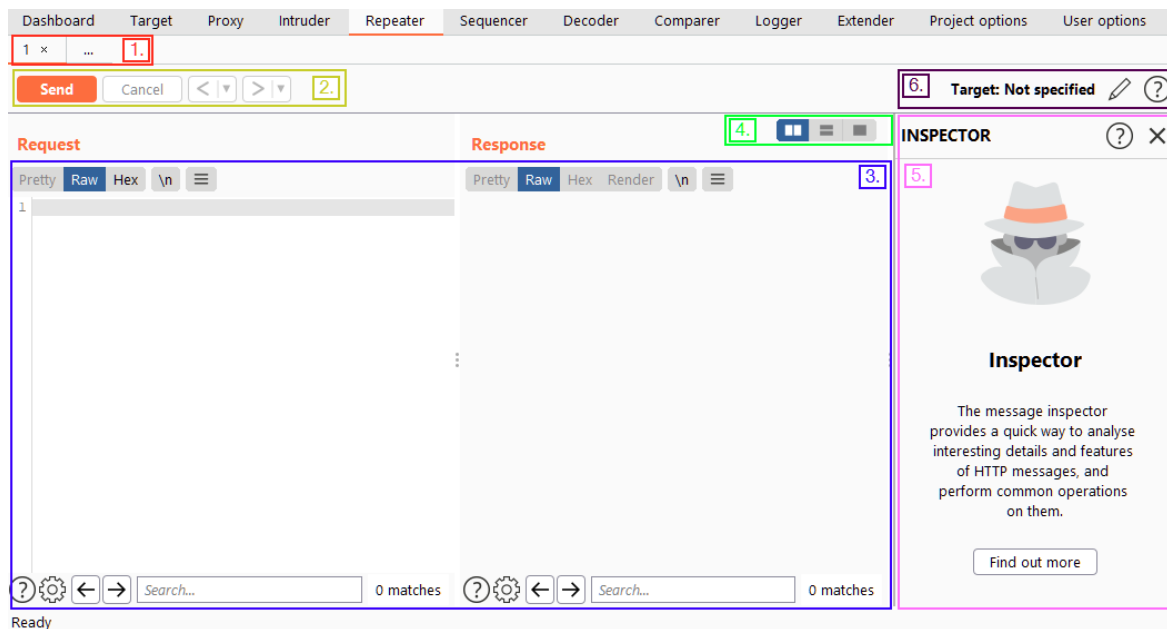
Avant de commencer à utiliser Repeater, il est utile d'avoir une bonne idée de ce qu'il fait.

En bref : Burp Suite Repeater nous permet de créer et/ou de relayer les requêtes interceptées vers une cible à volonté. En termes simples, cela signifie que nous pouvons prendre une requête capturée dans le proxy, la modifier et envoyer la même requête à plusieurs reprises autant de fois que nous le souhaitons. Alternativement, nous pourrions créer des requêtes à la main, un peu comme nous le ferions à partir de la CLI (command Line **Interface**) **C**, en utilisant un outil tel que CURL pour créer et envoyer des requêtes.

Fournissant une **GUI** agréable **graphique** Repeater **nous** pour écrire la charge utile de la requête et de nombreuses vues (y compris un moteur de rendu pour une vue graphique) de la réponse afin que nous puissions voir les résultats de notre travail manuel en action.

L'interface du répéteur peut être divisée en six sections principales -- un diagramme annoté se trouve sous les puces suivantes :

1. Tout en haut à gauche de l'onglet, nous avons une liste de demandes de répéteur. Nous pouvons avoir de nombreuses requêtes différentes passant par Repeater : chaque fois que nous envoyons une nouvelle requête à Repeater, elle apparaîtra ici.
2. Directement sous la liste des requêtes, nous avons les contrôles pour la requête en cours. Ceux-ci nous permettent d'envoyer une demande, d'annuler une demande suspendue et d'avancer/reculer dans l'historique des demandes.
3. Toujours sur le côté gauche de l'onglet, mais occupant la majeure partie de la fenêtre, nous avons la vue de la demande et de la réponse. Nous modifions la demande dans la vue Demande puis appuyons sur envoyer. La réponse s'affichera dans la vue Réponse.
4. Au-dessus de la section requête/réponse, sur le côté droit, se trouve un ensemble d'options nous permettant de modifier la disposition des vues de requête et de réponse. Par défaut, c'est généralement côte à côte (disposition horizontale, comme dans la capture d'écran) ; cependant, nous pouvons également choisir de les placer au-dessus/en dessous les uns des autres (disposition verticale) ou dans des onglets séparés (vue combinée).
5. Sur le côté droit de la fenêtre, nous avons l'inspecteur, qui nous permet de séparer les requêtes pour les analyser et les éditer de manière un peu plus intuitive qu'avec l'éditeur brut. Nous couvrirons cela dans une tâche ultérieure.
6. Enfin, au-dessus de l'inspecteur, nous avons notre cible. Il s'agit tout simplement de l'adresse IP ou du domaine auquel nous envoyons les requêtes. Lorsque nous envoyons des demandes à Repeater à partir d'autres parties de Burp Suite, celles-ci seront remplies automatiquement.



Ne vous inquiétez pas si cela n'a pas trop de sens pour le moment - vous aurez de nombreuses chances d'apprendre ce qu'il fait de première main dans les tâches à venir !

2.2 Utilisation de base

Nous savons à quoi ressemble l'interface maintenant, mais comment l'utiliser ?

Bien que nous puissions créer des demandes à la main, il serait beaucoup plus courant de simplement capturer une demande dans le proxy, puis de l'envoyer à Repeater pour modification/renvoi.

Avec une demande capturée dans le proxy, nous pouvons envoyer au répéteur soit en cliquant avec le bouton droit sur la demande et en choisissant "Envoyer au répéteur" ou en appuyant sur Ctrl + R.

En revenant à Repeater, nous pouvons voir que notre requête est désormais disponible :

The screenshot shows the Burp Suite Repeater tab. At the top, the target is set to `https://10-10-26-169.p.thmlabs.com`. Below the target bar, there are buttons for `Send`, `Cancel`, and navigation arrows. The `Request` section on the left shows a prepared HTTP GET request with the following details:

- Method: GET / HTTP/1.1
- Host: 10-10-26-169.p.thmlabs.com
- User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101 Firefox/91.0
- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
- Accept-Language: en-GB,en;q=0.5
- Accept-Encoding: gzip, deflate
- Referer: https://tryhackme.com/
- Dnt: 1
- Upgrade-Insecure-Requests: 1
- Sec-Fetch-Dest: document
- Sec-Fetch-Mode: navigate
- Sec-Fetch-Site: cross-site
- Sec-Fetch-User: ?1
- Sec-Gpc: 1
- Cache-Control: max-age=0
- Te: trailers
- Connection: close

The `Response` section is currently empty. The `Inspector` on the right shows the request attributes and headers, with 16 request headers listed.

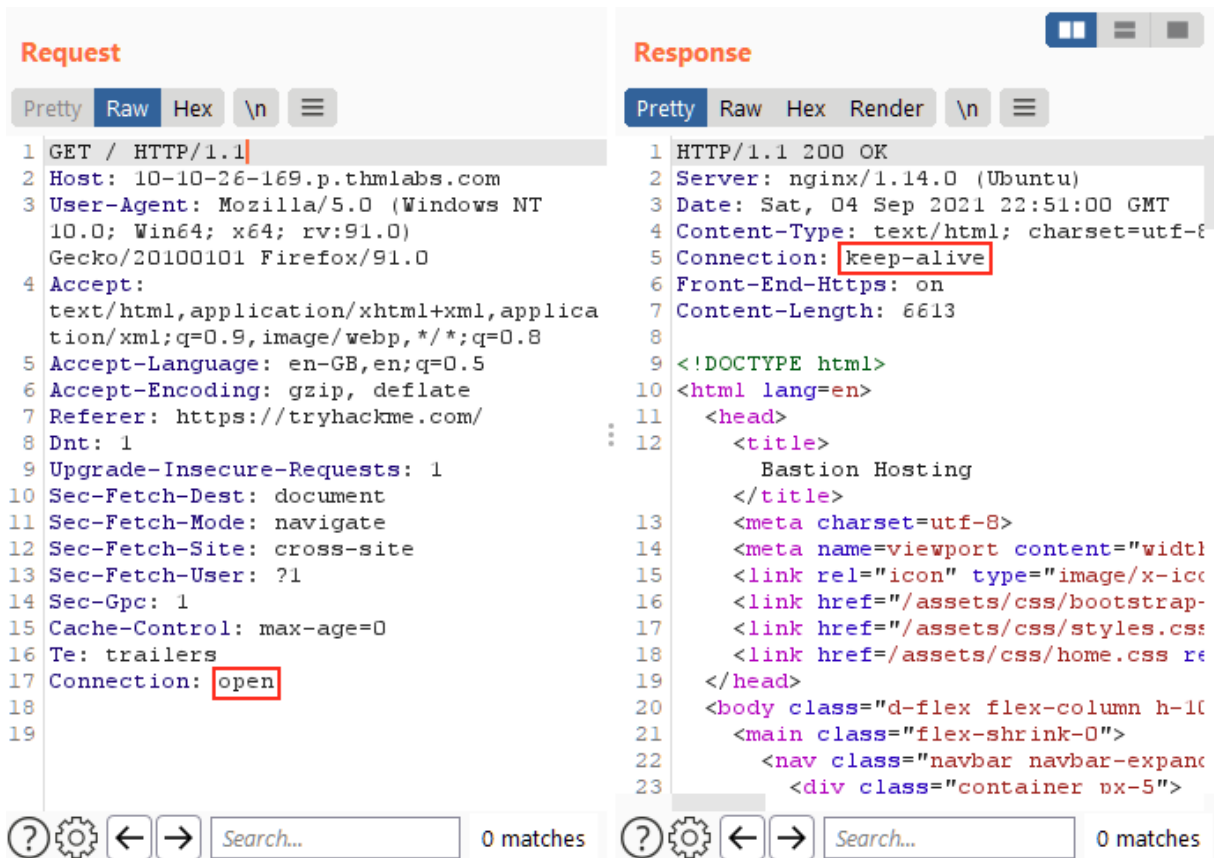
Les éléments cible et inspecteur affichent désormais également des informations ; cependant, nous n'avons pas encore de réponse. Lorsque nous cliquons sur le bouton "Envoyer", la section Réponse se remplit rapidement :

The screenshot shows the Burp Suite Repeater tab after the request has been sent. The `Response` section now contains the following details:

- Method: HTTP/1.1 200 OK
- Server: nginx/1.14.0 (Ubuntu)
- Date: Sat, 04 Sep 2021 22:50:16 GMT
- Content-Type: text/html; charset=utf-8
- Connection: close
- Front-End-Https: on
- Content-Length: 6613

The `Inspector` on the right now shows the response headers and the response body. The response body contains HTML code for a page titled "Bastion Hosting".

Si nous voulons changer quoi que ce soit à propos de la demande, nous pouvons simplement taper dans la fenêtre de demande et appuyer à nouveau sur "Envoyer" ; cela mettra à jour la réponse sur la droite. Par exemple, changer l'en-tête "Connexion" en open plutôt que close génère un en-tête de réponse "Connexion" avec une valeur de keep-alive:

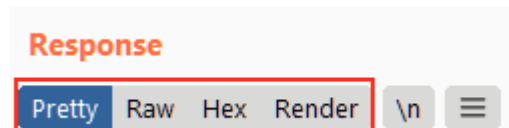


Nous pourrions alors également utiliser les boutons d'historique à droite du bouton Envoyer pour avancer et reculer dans notre historique des modifications.

2.3 View

Repeater nous propose différentes manières de présenter les réponses à nos demandes - celles-ci vont de la sortie hexadécimale jusqu'à une version entièrement rendue de la page.

Nous pouvons voir les options disponibles en regardant au-dessus de la boîte de réponse :



Nous avons quatre options d'affichage ici :

1. **Jolie** : il s'agit de l'option par défaut. Il prend la réponse brute et tente de l'embellir légèrement, ce qui la rend plus facile à lire.
2. **Raw** : la réponse pure et non embellie du serveur.
3. **Hex** : cette vue prend la réponse brute et nous en donne une vue en octets -- particulièrement utile si la réponse est un fichier binaire.
4. **Rendu** : la vue de rendu rend la page telle qu'elle apparaîtrait dans votre navigateur. Bien qu'il ne soit pas *extrêmement* utile étant donné que nous serions généralement intéressés par le code source lors de l'utilisation de Repeater, cela reste une astuce intéressante.

Dans la plupart des cas, l'option "Jolie" est parfaitement adéquate ; cependant, il vaut toujours la peine de savoir comment utiliser les trois autres options.

Juste à droite des boutons d'affichage se trouve le bouton "Afficher les caractères non imprimables" (\n). Ce bouton nous permet d'afficher des caractères qui n'apparaîtraient généralement pas dans les vues Jolie ou Brute. Par exemple, chaque ligne de la réponse se terminera par les caractères \r\n-- ceux-ci signifient un retour chariot suivi d'une nouvelle ligne et font partie de la façon dont les entêtes HTTP sont interprétés.

Bien qu'elle ne soit pas requise pour la plupart des tâches, cette option peut toujours être utile.

2.4 Inspector

À bien des égards, Inspector complète entièrement les champs de demande et de réponse de la fenêtre du répéteur. Si vous comprenez comment lire et modifier les requêtes HTTP, vous constaterez peut-être que vous utilisez rarement Inspector.

Cela dit, c'est un excellent moyen d'obtenir une répartition soignée des demandes et des réponses, ainsi que d'expérimenter pour voir comment les modifications apportées à l'aide de l'inspecteur de niveau supérieur affectent les versions brutes équivalentes.

L'inspecteur peut être utilisé dans le proxy ainsi que dans le répéteur. Dans les deux cas, il apparaît tout à droite de la fenêtre et nous donne une liste des composants de la requête et de la réponse :

INSPECTOR	?	X
Request Attributes		▼
Query Parameters (0)		▼
Body Parameters (0)		▼
Request Cookies (0)		▼
Request Headers (7)		▼
Response Headers (6)		▼

Parmi celles-ci, les sections de demande peuvent presque toujours être modifiées, ce qui nous permet d'ajouter, de modifier et de supprimer des éléments. Par exemple, dans la section Attributs de la requête, nous pouvons modifier les parties de la requête qui traitent de l'emplacement, de la méthode et du protocole ; par exemple, changer la ressource que nous cherchons à récupérer, modifier la requête de GET vers une autre méthode HTTP ou changer de protocole de HTTP/1 à HTTP/2 :

Request Attributes			^
Protocol			
			HTTP/1 HTTP/2
ATTRIBUTE	VALUE		
Method	GET		>
Path	/		>

Les autres sections disponibles pour l'affichage et/ou l'édition sont :

- **Paramètres de requête**, qui font référence aux données envoyées au serveur dans l'URL. Par exemple, dans une requête GET à `https://admin.tryhackme.com/?redirect=false`, il existe un paramètre de requête appelé "redirect" avec la valeur "false".
- **Body Parameters**, qui fait la même chose que Query Parameters, mais pour les requêtes POST. Tout ce que nous envoyons en tant que données dans une requête POST apparaîtra dans cette section, nous permettant une fois de plus de modifier les paramètres avant de renvoyer.
- **Les cookies de demande** contiennent, comme vous pouvez vous y attendre, une liste modifiable des cookies qui sont envoyés avec chaque demande.
- **Les en-têtes de demande** nous permettent de visualiser, d'accéder et de modifier (y compris l'ajout ou la suppression pur et simple) l'un des en-têtes envoyés avec nos demandes. Leur modification peut être très utile lorsque vous essayez de voir comment un serveur Web répondra à des en-têtes inattendus.
- **Les en-têtes de réponse** nous montrent les en-têtes que le serveur a renvoyés en réponse à notre requête. Ceux-ci ne peuvent pas être modifiés (car nous ne pouvons pas contrôler les en-têtes que le serveur nous renvoie !). Notez que cette section n'apparaîtra qu'après que nous ayons envoyé la demande et reçu une réponse.

Ces composants peuvent tous être trouvés sous forme de texte dans les sections de requête et de réponse ; cependant, il peut être agréable de les voir dans le format tabulaire proposé par Inspector. Il vaut la peine d'ajouter, de supprimer et de modifier des en-têtes dans Inspector pour avoir une idée de la façon dont la version brute change au fur et à mesure.

3. Burp Suite : Intruder

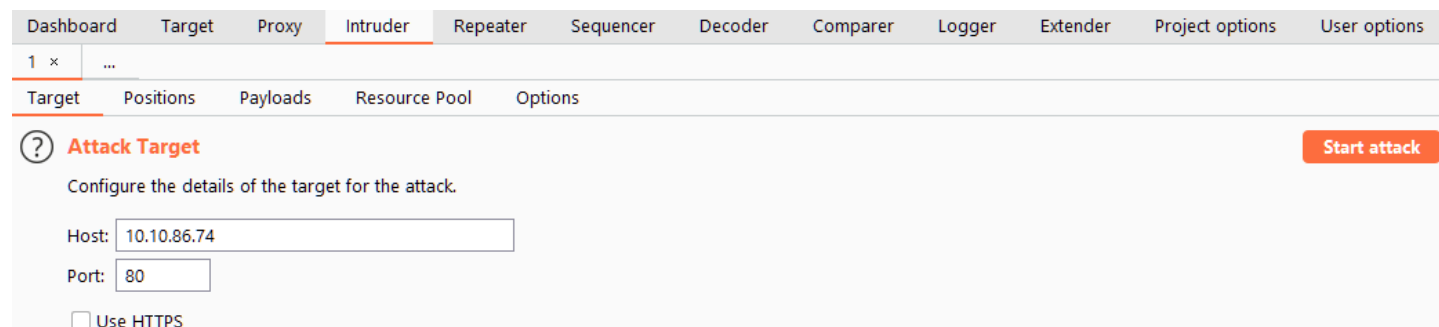
3.1 Qu'est-ce qu'Intruder?

Intruder est l'outil de fuzzing intégré à Burp Suite. Cela nous permet de prendre une requête (généralement capturée dans le proxy avant d'être transmise à Intruder) et de l'utiliser comme modèle pour envoyer automatiquement de nombreuses autres requêtes avec des valeurs légèrement modifiées. Par exemple, en capturant une demande contenant une tentative de connexion, nous pourrions alors configurer Intruder pour échanger les champs de nom d'utilisateur et de mot de passe contre des valeurs d'une liste de mots, nous permettant ainsi de forcer brutalement le formulaire de connexion. De même, nous pourrions transmettre une liste de mots fuzzing ^[1] et utiliser Intruder pour fuzzer les sous-répertoires, les points de terminaison ou les hôtes virtuels. Cette fonctionnalité est très similaire à celle fournie par les outils en ligne de commande tels que Wfuzz ou ffuf.

Bref, en tant que méthode d'automatisation des requêtes, Intruder est extrêmement puissant -- il n'y a qu'un seul problème : pour accéder à toute la vitesse d'Intruder, nous avons besoin de Burp Professional. Nous *pouvons* toujours utiliser Intruder avec Burp Community, mais le débit est fortement limité. Cette limitation de vitesse signifie que de nombreux pirates choisissent d'utiliser d'autres outils pour le fuzzing et le bruteforcing.

Limitations mises à part, Intruder est toujours très utile, il vaut donc la peine d'apprendre à l'utiliser correctement.

Jetons un coup d'œil à l'interface Intruder :



The screenshot shows the Burp Suite interface with the 'Intruder' tab selected. The 'Attack Target' sub-tab is active, displaying a form to configure the target for the attack. The form includes fields for 'Host' (10.10.86.74) and 'Port' (80), and a checkbox for 'Use HTTPS' which is currently unchecked. A 'Start attack' button is visible in the top right corner of the sub-tab.

La première vue que nous obtenons est une interface relativement clairessemée qui nous permet de choisir notre cible. En supposant que nous ayons envoyé une demande depuis le proxy (en utilisant Ctrl + I ou en cliquant avec le bouton droit de la souris et en sélectionnant "Envoyer à l'intrus"), cela devrait déjà être rempli pour nous.

Il existe quatre autres sous-onglets Intruder :

- **Positions** nous permettent de sélectionner un type d'attaque (nous les couvrirons dans une tâche à venir), ainsi que de configurer où dans le modèle de demande nous souhaitons insérer nos charges utiles.
- **Les charges utiles** nous permettent de sélectionner des valeurs à insérer dans chacune des positions que nous avons définies dans le sous-onglet précédent. Par exemple, nous pouvons choisir de charger des éléments à partir d'une liste de mots pour servir de charges utiles. La façon dont ceux-ci sont insérés dans le modèle dépend du type d'attaque que nous avons choisi dans l'onglet Positions. Il existe de nombreux types de charge utile parmi lesquels choisir (allant d'une simple liste de mots à des expressions régulières basées sur les réponses du serveur). Le sous-onglet Charges utiles nous permet également de modifier le comportement d'Intruder en ce qui concerne les charges utiles ; par exemple, nous pouvons

définir des règles de prétraitement à appliquer à chaque charge utile (par exemple, ajouter un préfixe ou un suffixe, faire correspondre et remplacer, ou ignorer si la charge utile correspond à une expression régulière définie).

- **Le pool de ressources** ne nous est pas particulièrement utile dans Burp Community. Cela nous permet de répartir nos ressources entre les tâches. Burp Pro nous permettrait d'exécuter divers types de tâches automatisées en arrière-plan, c'est là que nous souhaiterions peut-être allouer manuellement notre mémoire disponible et notre puissance de traitement entre ces tâches automatisées et Intruder. Sans accès à ces tâches automatisées, cela ne sert à rien de l'utiliser, nous n'y consacrerons donc pas beaucoup de temps.
- Comme avec la plupart des autres outils Burp, Intruder nous permet de configurer le comportement d'attaque dans le **Options** sous-onglet Les paramètres ici s'appliquent principalement à la façon dont Burp gère les résultats et à la façon dont Burp gère l'attaque elle-même. Par exemple, nous pouvons choisir de signaler les requêtes contenant des éléments de texte spécifiés ou de définir la manière dont Burp répond aux réponses de redirection (3xx).

Nous examinerons de plus près certains de ces sous-onglets dans les tâches à venir. Pour l'instant, apprenez simplement où en sont les choses dans l'interface.

1. Le fuzzing consiste à prendre un ensemble de données et à l'appliquer à un paramètre pour tester une fonctionnalité ou pour voir si quelque chose existe. Par exemple, nous pouvons choisir de "fuzzer les terminaux" dans une application Web ; cela impliquerait de prendre chaque mot dans une liste de mots et de l'ajouter à la fin d'une requête pour voir comment le serveur Web répond (par exemple `http://MACHINE_IP/WORD_GOES_HERE`).

3.2 Positions

Lorsque nous cherchons à effectuer une attaque avec Intruder, la première chose que nous devons faire est de regarder les *positions*. Les positions indiquent à Intruder où insérer les charges utiles (que nous examinerons dans les tâches à venir).

Passons au sous-onglet Postes :

Target

Positions

Payloads

Options

?

Payload Positions

Start attack

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type:

Sniper

1 POST /support/login/ HTTP/1.1

2 Host: 10.10.86.74

3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

5 Accept-Language: en-US,en;q=0.5

6 Accept-Encoding: gzip, deflate

7 Content-Type: application/x-www-form-urlencoded

8 Content-Length: 37

9 Origin: http://10.10.86.74

10 Connection: close

11 Referer: http://10.10.86.74/support/login/

12 Upgrade-Insecure-Requests: 1

13

14 username=\$pentester\$&password=\$Exp101ted\$

Add \$

Clear \$

Auto \$

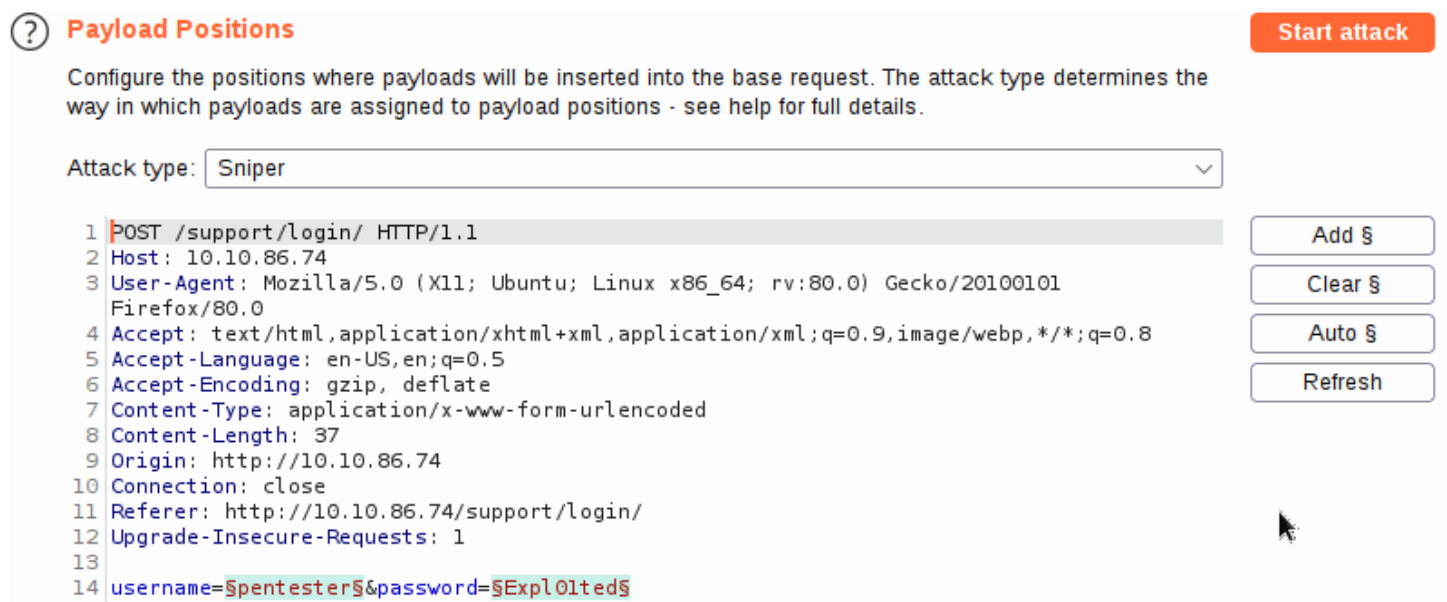
Refresh

Notez que Burp tentera de déterminer les endroits les plus probables où nous pourrions souhaiter insérer automatiquement une charge utile - ceux-ci sont surlignés en vert et entourés de silcrows (§).

Sur le côté droit de l'interface, nous avons les boutons "Ajouter §", "Effacer §" et "Auto §":

- **Ajouter** nous permet de définir de nouveaux postes en les mettant en surbrillance dans l'éditeur et en cliquant sur le bouton.
- **Effacer** supprime toutes les positions définies, nous laissant avec une toile vierge pour définir la nôtre.
- **Auto** tente de sélectionner automatiquement les positions les plus probables ; ceci est utile si nous avons effacé les positions par défaut et que nous voulons les récupérer.

Voici un GIF illustrant le processus d'ajout, d'effacement et de resélections automatique de postes :



Passons au sous-onglet "Positions" et regardons dans le menu déroulant "Types d'attaques".

Quatre types d'attaques sont disponibles :

- Sniper
- Béliet
- Fourche
- Bombe à fragmentation

Nous examinerons chacun d'eux à tour de rôle.

3.3 Sniper

Sniper est le premier type d'attaque et le plus courant.

Lors d'une attaque de Sniper, nous fournissons un ensemble de charges utiles. Par exemple, il peut s'agir d'un seul fichier contenant une liste de mots ou une plage de nombres. À partir de maintenant, nous ferons référence à une liste d'éléments à insérer dans les demandes en utilisant la terminologie Burp Suite d'un "ensemble de charge utile". Intruder prendra chaque charge utile dans un ensemble de charges utiles et la placera dans chaque position définie à tour de rôle.

Jetez un œil à notre exemple de modèle d'avant :

Exemples de postes

```
POST /support/login/ HTTP/1.1
Hôte : MACHINE_IP
Agent utilisateur : Mozilla/5.0 (X11 ; Ubuntu ; Linux x86_64 ; rv:80.0) Gecko/20100101 Firefox/80.0
Accepter : text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language : en-US,en;q=0.5
Accepter-Encodage : gzip, dégonfler
Type de contenu : application/x-www-form-urlencoded
Longueur du contenu : 37
Origine : http:// IP_MACHINE
Connexion : fermer
Réfèrent : http:// MACHINE_IP /support/login/
Demandes de mise à niveau non sécurisées : 1
utilisateur= $pentester$ &mot de passe= $Expl01ted$
```

Deux positions sont définies ici, ciblant les paramètres **username** et **password** corporels.

Lors d'une attaque de tireur d'élite, Intruder prendra chaque position et y substituera chaque charge utile à tour de rôle.

Par exemple, supposons que nous ayons une liste de mots contenant trois mots : burp, suite, et intruder.

Avec les deux positions que nous avons ci-dessus, Intruder utiliserait ces mots pour faire **six** requêtes :

Numéro de demande	Corps de la requête
1	username=burp&password=Expl01ted
2	username=suite&password=Expl01ted
3	username=intruder&password=Expl01ted
4	username=pentester&password=burp
5	username=pentester&password=suite
6	username=pentester&password=intruder

Remarquez comment Intruder commence par la première position (username) et essaie chacune de nos charges utiles, puis passe à la deuxième position et essaie à nouveau les mêmes charges utiles. Nous pouvons calculer le nombre de requêtes que fera Intruder Sniper comme $\text{requests} = \text{numberOfWords} * \text{numberOfPositions}$. Cette qualité rend Sniper très bon pour les attaques à position unique (par exemple, un mot de passe force brute si nous connaissons le nom d'utilisateur ou le *fuzzing pour les API endpoint* s).

3.4 Bélier

Ensuite, jetons un coup d'œil au *Battering Ram Attack*.

Comme Sniper, Battering ram prend *un* ensemble de charges utiles (par exemple une liste de mots). *Contrairement* au Sniper, le Bélier met la même charge utile dans *chaque* position plutôt que dans chaque position à tour de rôle.

Utilisons la même liste de mots et la même demande d'exemple que dans la dernière tâche pour illustrer cela.

Exemples de postes

```
POST /support/login/ HTTP/1.1
Hôte : MACHINE_IP
Agent utilisateur : Mozilla/5.0 (X11 ; Ubuntu ; Linux x86_64 ; rv:80.0) Gecko/20100101 Firefox/80.0
Accepter : text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language : en-US,en;q=0.5
Accepter-Encodage : gzip, dégonfler
Type de contenu : application/x-www-form-urlencoded
Longueur du contenu : 37
Origine : http:// IP_MACHINE
Connexion : fermer
Réfèrent : http:// MACHINE_IP /support/login/
Demandes de mise à niveau non sécurisées : 1

utilisateur= $pentester$ &mot de passe= $Expl01ted$
```

Si nous utilisons Battering ram pour attaquer cela, Intruder prendra chaque charge utile et la remplacera dans chaque position *à la fois*.

Avec les deux positions que nous avons ci-dessus, Intruder utiliserait les trois mots d'avant (burp, suite, et intruder) pour faire *trois* requêtes :

Numéro de demande	Corps de la requête
1	username=burp&password=burp
2	username=suite&password=suite
3	username=intruder&password=intruder

Comme on peut le voir dans le tableau, chaque élément de notre liste de charges utiles est placé dans *chaque* position pour chaque requête. Fidèle à son nom, Battering ram lance simplement des charges utiles sur la cible pour voir ce qui colle.

3.5 Fourchette

Deux de moins, encore deux !

Après Sniper, Pitchfork est le type d'attaque que vous êtes le plus susceptible d'utiliser. Il peut être utile de penser à Pitchfork comme si de nombreux tireurs d'élite s'exécutaient simultanément. Là où Sniper utilise *un* ensemble de charges utiles (qu'il utilise simultanément sur toutes les positions), Pitchfork utilise un ensemble de charges utiles par position (jusqu'à un maximum de 20) et les parcourt tous en même temps.

Ce type d'attaque peut prendre un peu de temps pour comprendre, alors utilisons notre exemple de force brute d'avant, mais cette fois nous avons besoin de deux listes de mots :

- Notre première liste de mots sera les noms d'utilisateur. Il contient trois entrées : joel, harriet, alex.
- Disons que Joel, Harriet et Alex ont eu leur mot de passe divulgué : nous savons que le mot de passe de Joel est J03l, le mot de passe d'Harriet est Emma1815, et le mot de passe d'Alex est Sk1ll.

Nous pouvons utiliser ces deux listes pour effectuer une attaque pitchfork sur le formulaire de connexion d'avant. Le processus de réalisation de cette attaque ne sera pas couvert dans cette tâche, mais vous aurez de nombreuses occasions d'effectuer des attaques comme celle-ci plus tard !

Lors de l'utilisation d'Intruder en mode fourche, les requêtes effectuées ressembleraient à ceci :

Numéro de demande	Corps de la requête
1	username=joel&password=J03l
2	username=harriet&password=Emma1815
3	username=alex&password=Sk1ll

Voyez comment Pitchfork prend le premier élément de chaque liste et le place dans la requête, un par position ? Il répète ensuite cela pour la requête suivante : en prenant le deuxième élément de chaque liste et en le remplaçant dans le modèle. L'intrus continuera à le faire jusqu'à ce qu'une (ou toutes) des listes soit épuisée. Idéalement, nos ensembles de charges utiles devraient être de longueurs identiques lorsque vous travaillez dans Pitchfork, car Intruder arrêtera les tests dès que l'une des listes sera complète. Par exemple, si nous avons deux listes, une de 100 lignes et une de 90 lignes, Intruder ne fera que 90 requêtes et les dix derniers éléments de la première liste ne seront pas testés.

Ce type d'attaque est exceptionnellement utile pour former des choses comme les attaques de credential stuffing (nous venons d'en rencontrer une version à petite échelle). Nous les examinerons plus en détail plus tard dans la salle.

3.6 Bombe

Enfin, nous arrivons au dernier type d'attaque d'Intruder : la bombe à fragmentation.

Comme Pitchfork, la bombe à fragmentation nous permet de choisir plusieurs ensembles de charges utiles : un par position, jusqu'à un maximum de 20 ; cependant, alors que Pitchfork parcourt simultanément chaque ensemble de charges utiles, la bombe à fragmentation parcourt chaque ensemble de charges utiles individuellement, en s'assurant que chaque combinaison possible de charges utiles est testée.

Encore une fois, la meilleure façon de visualiser cela est avec un exemple.

Utilisons les mêmes listes de mots qu'avant :

- Noms d'utilisateur : joel, harriet, alex.
- Mots de passe : J03l, Emma1815, Sk1ll.

Mais, cette fois, supposons que nous ne sachions pas quel mot de passe appartient à quel utilisateur. Nous avons trois utilisateurs et trois mots de passe, mais nous ne savons pas comment les faire correspondre. Dans ce cas, nous utiliserions une attaque à la bombe à fragmentation ; cela va essayer *toutes les* combinaisons de valeurs. La table de demande pour nos positions de nom d'utilisateur et de mot de passe ressemble à ceci :

Numéro de demande	Corps de la requête
1	username=joel&password=J03l
2	username=harriet&password=J03l
3	username=alex&password=J03l
4	username=joel&password=Emma1815
5	username=harriet&password=Emma1815
6	username=alex&password=Emma1815
7	username=joel&password=Sk1ll
8	username=harriet&password=Sk1ll
9	username=alex&password=Sk1ll

Cluster Bomb parcourra chaque combinaison des ensembles de charges utiles fournis pour s'assurer que chaque possibilité a été testée. Ce type d'attaque peut créer une *énorme* quantité de trafic (égale *au nombre de lignes dans chaque ensemble de données utiles multipliées ensemble*), alors soyez prudent ! De même, lorsque vous utilisez Burp Community et sa limitation de débit Intruder, sachez qu'une attaque à la bombe à fragmentation avec n'importe quel ensemble de charges utiles de taille moyenne prendra un temps incroyablement long.

Cela dit, il s'agit d'un autre type d'attaque extrêmement utile pour tout type de force brute d'identification où un nom d'utilisateur n'est pas connu.

3.7 Intruder Payloads

C'était beaucoup de théorie, alors félicitations pour l'avoir lu ! Il y aura beaucoup de travaux pratiques dans les tâches à venir, mais d'abord, il est impératif que nous comprenions comment créer, attribuer et utiliser des charges utiles.

Passez au sous-onglet "Payloads" ; celui-ci est divisé en quatre sections :

- La **Ensembles de charge utile** nous permet de choisir la position pour laquelle nous voulons configurer un ensemble ainsi que le type de charge utile que nous aimerions utiliser.
 - Lorsque nous utilisons un type d'attaque qui n'autorise qu'un seul ensemble de charges utiles (c.-à-d. Sniper ou Battering Ram), le menu déroulant pour "Ensemble de charges utiles" n'aura qu'une seule option, quel que soit le nombre de positions que nous avons définies.
 - Si nous utilisons l'un des types d'attaques qui utilisent plusieurs ensembles de charges utiles (c'est-à-dire Pitchfork ou Cluster Bomb), alors il y aura un élément dans la liste déroulante pour chaque position.

Remarque : Les positions multiples doivent être lues de haut en bas, puis de gauche à droite lorsqu'elles se voient attribuer des numéros dans la liste déroulante "Ensemble de charge utile". Par exemple, avec deux positions (username=\$pentester&password=\$Exp101ted\$), le premier élément de la liste déroulante de l'ensemble de données utiles ferait référence au champ du nom d'utilisateur et le second au champ du mot de passe.

La deuxième liste déroulante de cette section nous permet de sélectionner un "type de charge utile". Par défaut, il s'agit d'une "liste simple" -- qui, comme son nom l'indique, nous permet de charger une liste de mots à utiliser. Il existe de nombreux autres types de charge utile disponibles. Parmi les plus courants, citons : Recursive Grep, Numbers, et Username generator. Il vaut la peine de parcourir cette liste pour avoir une idée du large éventail d'options disponibles.

- Les **options de charge utile** diffèrent selon le type de charge utile que nous sélectionnons pour l'ensemble de charge utile actuel. Par exemple, un type de charge utile « Liste simple » nous donnera une boîte pour ajouter et supprimer des charges utiles vers et depuis l'ensemble :

? **Payload Options [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

Paste Load ... Remove Clear

joel
harriet
alex

Add

Add from list ... [Pro version only]

- Nous pouvons le faire manuellement en utilisant la zone de texte "Ajouter", coller des lignes avec "Coller" ou "Charger..." à partir d'un fichier. Le bouton "Supprimer" supprime *uniquement* . Le bouton "Effacer" efface toute la liste. Soyez averti : charger des listes extrêmement volumineuses ici peut faire planter Burp !
En revanche, les options pour un NumbersLe type de charge utile nous permet de modifier des options telles que la plage de numéros utilisée et la base avec laquelle nous travaillons.
- **Le traitement de la charge utile** nous permet de définir des règles à appliquer à chaque charge utile de l'ensemble avant d'être envoyées à la cible. Par exemple, nous pourrions mettre en majuscule chaque mot ou ignorer la charge utile si elle correspond à une expression régulière. Vous n'utiliserez peut-être pas cette section particulièrement régulièrement, mais vous l'apprécierez certainement lorsque vous en aurez *besoin* !
- Enfin, nous avons la **Payload Encoding** . Cette section nous permet de remplacer les options de codage d'URL par défaut qui sont appliquées automatiquement pour permettre la transmission en toute sécurité de notre charge utile. Parfois, il peut être avantageux de *ne pas* coder en URL ces caractères "non sécurisés" standard, c'est là qu'intervient cette section. Nous pouvons soit ajuster la liste des caractères à encoder, soit décocher carrément la case "encoder ces caractères en URL".

Lorsqu'elles sont combinées, ces sections nous permettent d'adapter parfaitement nos ensembles de charges utiles à toute attaque que nous souhaitons mener.

4. Burp Suite : Other Modules

4.1 Présentation Décodeur

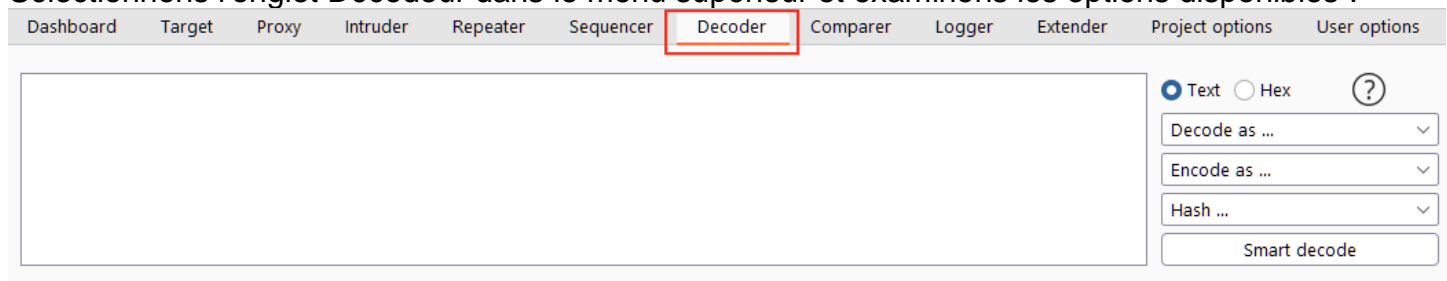
Outre les célèbres [Repeater](#) et [Intruder](#), Burp Suite intègre également plusieurs modules un peu plus obscurs : c'est ce que nous allons couvrir dans cette partie.

Plus précisément, nous examinerons les *Décodeur*, *Compareur* et *Séquenceur*. Ceux-ci nous permettent de : travailler avec du texte encodé ; comparer des ensembles de texte ; et analyser le caractère aléatoire des jetons capturés, respectivement. Être capable d'effectuer ces tâches relativement simples directement dans Burp Suite peut faire gagner beaucoup de temps, il vaut donc bien le temps passé à apprendre à utiliser ces modules efficacement.

Sans plus tarder, sautons et commençons à regarder Decoder !

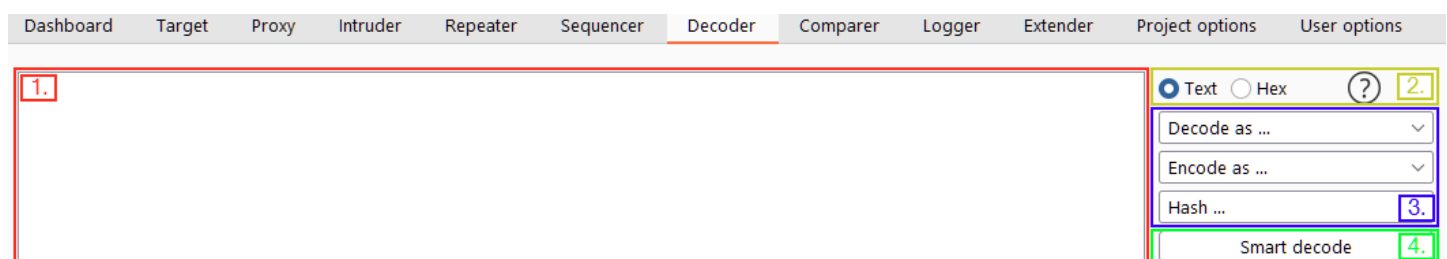
Le module Burp Suite Decoder nous permet de manipuler des données. Comme son nom l'indique, nous pouvons *décoder* les informations que nous capturons lors d'une attaque, mais nous pouvons également *encoder* nos propres données, prêtes à être envoyées à la cible. Decoder nous permet également de créer *hashsums* de données, ainsi que de fournir une fonctionnalité Smart Decode qui tente de décoder les données fournies de manière récursive jusqu'à ce qu'elles redeviennent du texte en clair (comme la fonction "Magic" de [Cyberchef](#)).

Sélectionnons l'onglet Décodeur dans le menu supérieur et examinons les options disponibles :



Cette interface nous offre un certain nombre d'options.

1. La boîte de gauche est l'endroit où nous collerions ou taperions le texte à encoder ou à décoder. Comme avec la plupart des autres modules de Burp Suite, nous pouvons également envoyer des données ici à partir d'autres sections du framework en cliquant avec le bouton droit de la souris et en choisissant *Envoyer au décodeur*.
2. Nous avons la possibilité de choisir entre traiter l'entrée comme du texte ou des valeurs d'octets hexadécimaux en haut de la liste à droite.
3. Plus bas dans la liste, nous avons des menus déroulants pour *encoder*, *décoder* ou *hacher* l'entrée.
4. Enfin, nous avons la fonction "Smart Decode", qui tente de décoder automatiquement l'entrée.



Au fur et à mesure que nous ajoutons des données dans le champ d'entrée, l'interface se dupliquera pour contenir la sortie de notre transformation. On peut alors choisir d'opérer dessus en utilisant les mêmes options :

Nous examinerons les transformations disponibles dans les tâches à venir !

4.2 décodeurs Encodage/Décodage

Méthodes de décodage/encodage :

Examinons de plus près les options d'encodage et de décodage manuels. Ce sont les mêmes que l'on choisisse le menu de décodage ou d'encodage :

- **Plain** : le texte en clair est ce que nous avons avant d'effectuer toute transformation.
- **URL** : Le codage d'URL est utilisé pour sécuriser le transfert des données dans l'URL d'un site Web demande. Il s'agit d'échanger des caractères contre leur caractère ASCII code au format hexadécimal, précédé d'un symbole de pourcentage (%). L'encodage d'URL est une méthode extrêmement utile à connaître pour tout type de test d'application Web. Par exemple, encodons le caractère barre oblique (/). Le [caractère ASCII](#) pour une barre oblique est 47. Il s'agit de "2F" en hexadécimal, ce qui rend l'URL encodée en barre oblique. %2F. Nous pouvons confirmer cela avec Decoder en tapant une barre oblique dans la zone de saisie, puis en sélectionnant Encode as-> URL:

HTML : l'encodage de texte en tant qu'entités HTML implique le remplacement de caractères spéciaux par une esperluette (&) suivi d'un nombre hexadécimal ou d'une référence au caractère échappé, puis d'un point-virgule (;). Par exemple, un guillemet a sa propre référence : ". Lorsqu'il est inséré dans une page Web, il est remplacé par un guillemet double ("). Cette méthode d'encodage autorise les caractères spéciaux dans le langage HTML pour être rendu en toute sécurité dans les pages HTML et a l'avantage supplémentaire d'utiliser pour empêcher les attaques telles que [XSS](#) (Cross-Site Scripting).

Lorsque nous utilisons le HTML option dans Decoder, nous pouvons encoder n'importe quel caractère car son HTML s'est échappé formater ou décoder les entités HTML capturées. Par exemple, pour décoder le guillemet que nous avons regardé auparavant, nous tapons la variante codée puis choisissons Decode as-> HTML:

The screenshot shows the Decoder tool interface. The input field contains the text """. The output field is empty. The 'Text' radio button is selected, and the 'Decode as' dropdown is set to 'HTML'. The 'Encode as' dropdown is also set to 'HTML'. The 'Hash' dropdown is set to 'MD5'. The 'Smart decode' button is visible.

- **Base64** : Une autre méthode d'encodage largement utilisée, base64 est utilisée pour encoder toutes les données dans un format compatible ASCII. Il a été conçu pour prendre des données binaires (par exemple des images, médias, programmes) et l'encoder dans un format qui conviendrait pour transférer sur pratiquement n'importe quel support. Comment cela fonctionne sous le capot n'est pas important à ce stade ; cependant, si vous êtes intéressé, vous peut lire les maths derrière [ici](#).
- **Hexagone ASCII** : Cette option convertit les données entre représentation ASCII et hexadécimal représentation. Par exemple, le mot "ASCII" peut être converti en nombre hexadécimal "4153434949". Chaque lettre des données d'origine est prise individuellement et convertis de la représentation numérique ASCII en hexadécimal. Décimal [code de caractère](#) de 65. En hexadécimal, c'est 41. De même, la lettre "S" peut être convertie en hexadécimal 53, et ainsi de suite.
- **Hex , Octal et Binaire** : Ces les méthodes de codage s'appliquent toutes uniquement aux entrées numériques. Ils convertissent entre décimal, hexadécimal, octal (base huit) et binaire.
- **Gzip** : Gzip permet de *compresser* Les données. Il est largement utilisé pour réduire la taille des fichiers et des pages avant ils sont envoyés à votre navigateur. Des pages plus petites signifient des temps de chargement plus rapides, ce qui est hautement souhaitable pour les développeurs qui cherchent à augmenter leur score SEO et à éviter d'ennuyer leurs clients. Le décodeur nous permet d'encoder manuellement et décoder les données gzip, bien que cela puisse être difficile à traiter car il est souvent ASCII/Unicode non valide. Par exemple :

The screenshot shows the Decoder tool interface. The input field contains the text "Testing Gzip". The output field contains the compressed data "H4sI0000I-ÉikwpTÊ:0000▲". The 'Text' radio button is selected, and the 'Decode as' dropdown is set to 'Gzip'. The 'Encode as' dropdown is also set to 'Gzip'. The 'Hash' dropdown is set to 'MD5'. The 'Smart decode' button is visible.

Nous pouvons superposer n'importe lequel d'entre eux. Par exemple, nous pourrions prendre une phrase ("Burp Suite Decoder"), la convertir en ASCII Hex, puis en octal :



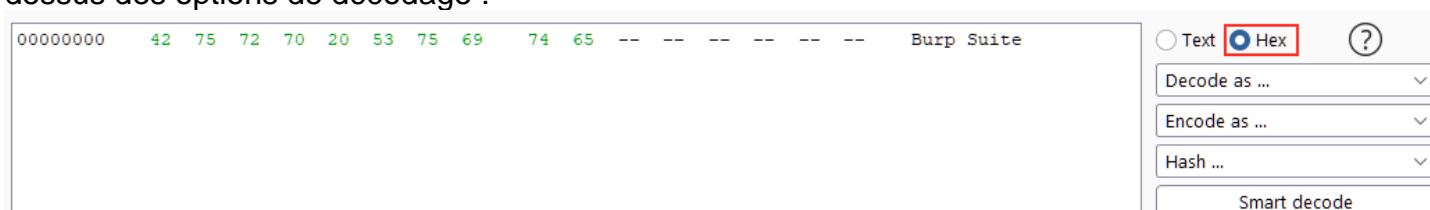
The image shows two identical, empty Burp Suite decoder panels. Each panel consists of a large text input area on the left and a control panel on the right. The control panel includes radio buttons for 'Text' (selected) and 'Hex', a help icon (?), and four dropdown menus labeled 'Decode as ...', 'Encode as ...', 'Hash ...', and 'Smart decode'.

Lorsqu'elles sont combinées, ces méthodes nous donnent un grand contrôle sur les données que nous encodons ou décodons.

Comme vous l'avez peut-être remarqué dans les exemples, chaque méthode d'encodage/décodage est codée par couleur pour nous permettre de voir rapidement et facilement quelle transformation a été appliquée.

Format hexadécimal :

La saisie de données au format ASCII est excellente, mais nous devons parfois pouvoir modifier notre entrée octet par octet. Pour cela, nous pouvons utiliser "Hex View" en choisissant "Hex" au-dessus des options de décodage :



The image shows a Burp Suite decoder panel with the 'Hex' radio button selected and highlighted by a red box. The text input area displays the hex string '00000000 42 75 72 70 20 53 75 69 74 65 -- -- -- -- -- -- Burp Suite'. The control panel on the right is identical to the one in the previous image, with 'Text' and 'Hex' radio buttons, a help icon (?), and four dropdown menus.

Ce paramètre nous permet d'afficher et de modifier notre texte au format octet hexadécimal - une astuce très utile si nous travaillons avec des fichiers binaires ou d'autres données non ASCII.

Décodage intelligent :

Enfin, jetons un coup d'œil à l'option "Smart Decode". Cette fonctionnalité de Decoder tente de décoder automatiquement le texte encodé. Par exemple, `Burp Suite`, est automatiquement reconnu comme étant encodé en HTML et est automatiquement décodé en conséquence :

Cette fonctionnalité est loin d'être parfaite, mais elle peut être très utile pour décoder rapidement des morceaux de données inconnues.

4.3 Hachage

La théorie :

Le hachage est un processus unidirectionnel utilisé pour transformer les données en une signature unique. Pour être un algorithme de hachage, la sortie résultante doit être impossible à inverser. Un bon algorithme de hachage garantira que chaque élément de données saisi aura un hachage complètement unique. Par exemple, l'utilisation de l' [algorithme MD5](#) pour générer une somme de hachage pour le texte "MD5sum" renvoie 4ae1a02de5bd02a5515f583f4fca5e8c. L'utilisation du même algorithme pour générer une somme de hachage pour "MD5SUM" donne un hachage complètement différent, malgré les similitudes de l'entrée : 13b436b09172400c9eb2f69fbd20adad. Pour cette raison, les hachages sont fréquemment utilisés pour vérifier l'intégrité des fichiers et des documents, car même une très petite modification du fichier entraînera une modification significative de la somme de hachage.

Remarque : *l'algorithme MD5 est obsolète et ne doit pas être utilisé pour les applications modernes.*

De même, les hachages sont également utilisés pour stocker en toute sécurité les mots de passe car (en raison du processus de hachage unidirectionnel, ce qui signifie que les hachages ne peuvent jamais être inversés), les mots de passe seront (relativement) sécurisés même en cas de fuite de la base de données. Lorsqu'un utilisateur crée un mot de passe, il est haché et stocké par l'application. Lorsque l'utilisateur essaie de se connecter, l'application hache alors le mot de passe qu'il soumet et le compare au hachage stocké ; si les hachages correspondent, alors le mot de passe était correct. Lors de l'utilisation de cette méthodologie, une application n'a jamais à stocker le mot de passe d'origine (en clair).

Hachage dans le décodeur :

Decoder nous permet de générer des hashsums pour les données directement dans Burp Suite ; cela fonctionne à peu près de la même manière que les options d'encodage/décodage que nous avons vues dans la tâche précédente. Plus précisément, nous cliquons sur le menu déroulant "Hash" et sélectionnons un algorithme dans la liste :

Hash ...

1.0.10118.3.0.55

1.2.804.2.1.1.1.1.2.2.1

1.2.804.2.1.1.1.1.2.2.2

1.2.804.2.1.1.1.1.2.2.3

2.16.840.1.101.3.4.2.10

2.16.840.1.101.3.4.2.11

2.16.840.1.101.3.4.2.12

2.16.840.1.101.3.4.2.7

2.16.840.1.101.3.4.2.8

Remarque : il s'agit d'une liste nettement plus longue qu'avec les algorithmes d'encodage/décodage -- cela vaut la peine de faire défiler la liste pour voir les nombreux algorithmes de hachage disponibles.

Poursuivant notre exemple précédent, entrons "MD5sum" dans la zone de saisie, puis faites défiler la liste jusqu'à ce que nous trouvions "MD5". L'application de ceci nous envoie automatiquement dans la vue Hex :

MD5sum

Text

Hex

?

Decode as ...

Encode as ...

Hash ...

Smart decode

Text

Hex

Decode as ...

Encode as ...

Hash ...

Smart decode

00000000 4a e1 a0 2d e5 bd 02 a5 51 5f 58 3f 4f ca 5e 8c JÁ -À-Œ_Q_X?OE^□

En effet, la sortie d'un algorithme de hachage ne renvoie pas de texte ASCII/Unicode pur. En tant que tel, il est courant de prendre la sortie résultante de l'algorithme et de la transformer en une chaîne hexadécimale ; c'est la forme de "hachage" que vous connaissez peut-être.

Terminons cela en appliquant un encodage "ASCII Hex" à la somme de hachage pour créer la chaîne hexadécimale soignée de notre exemple initial.Le processus complet peut être vu ici:

Text

Hex

?

Decode as ...

Encode as ...

Hash ...

Smart decode

Text

Hex

Decode as ...

Encode as ...

Hash ...

Smart decode

4.4 Présentation Comparer

Comme son nom l'indique, *Comparer* nous permet de comparer deux données, soit par mots ASCII, soit par octets.

Commençons par jeter un œil à l'interface :

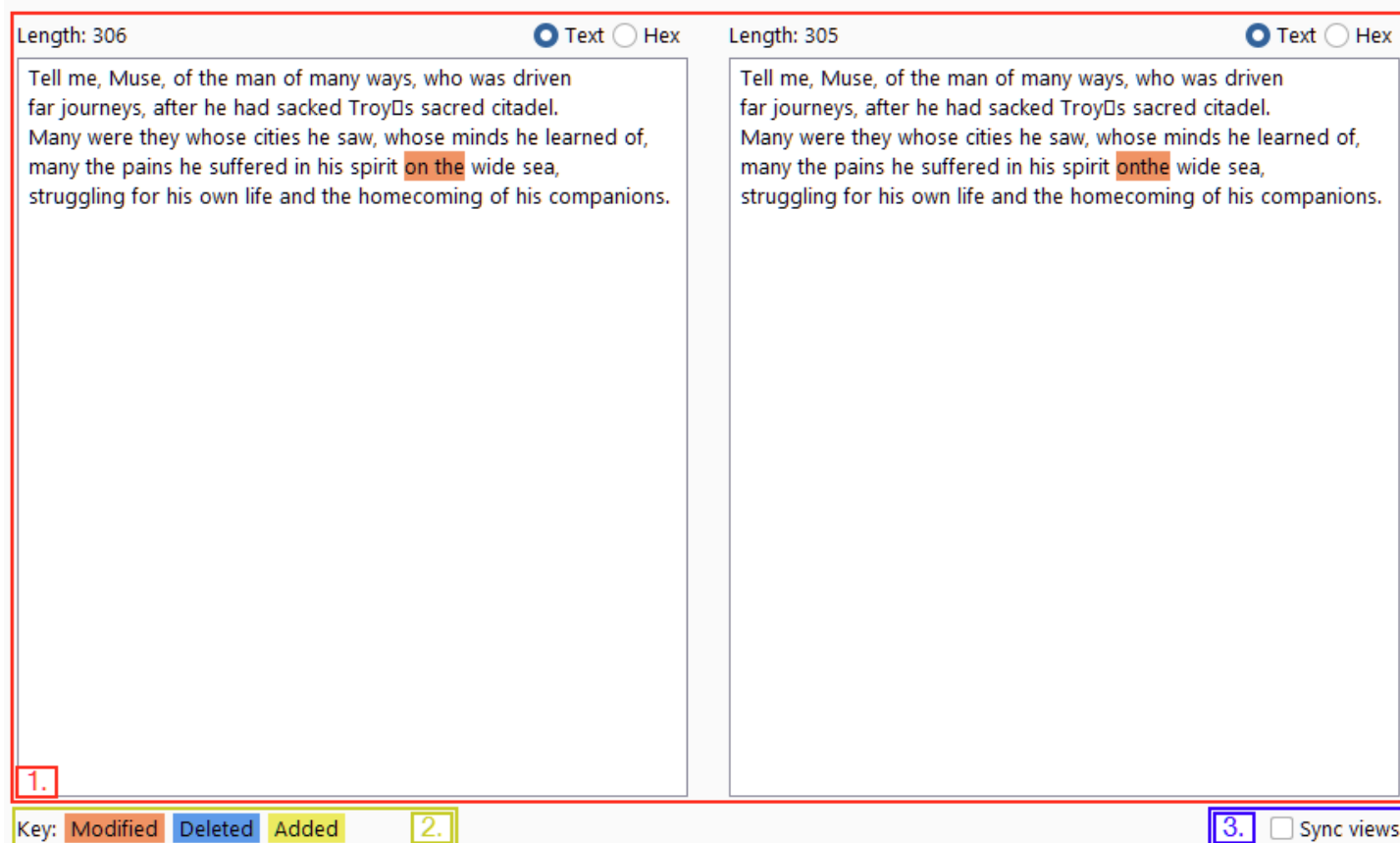
The screenshot shows the 'Comparer' tab in a web application. At the top is a navigation bar with tabs: Dashboard, Target, Proxy, Intruder, Repeater, Sequencer, Decoder, Comparer (active), Logger, Extender, Project options, and User options. Below the navigation bar, the 'Comparer' section has a title and a description: 'This function lets you do a word- or byte-level comparison between different data. You can load, paste, or send data here from other tools and then select the comparison you want to perform.' The main area contains two identical data entry sections, 'Select item 1:' and 'Select item 2:'. Each section has a table with three columns: '#', 'Length', and 'Data'. The first table is highlighted with a red border and a red box labeled '1.' in the bottom left corner. The second table is also highlighted with a red border. To the right of the first table is a vertical stack of buttons: 'Paste' (highlighted with a yellow box and label '2.'), 'Load', 'Remove', and 'Clear'. To the right of the second table is a vertical stack of buttons: 'Compare ...' (highlighted with a blue box and label '3.'), 'Words', and 'Bytes'.

Cette interface peut être divisée en trois parties principales :

1. Sur la gauche, nous avons les éléments comparés. Lorsque nous chargeons des données dans Comparer, elles apparaissent sous forme de lignes dans ces tables ; nous sélectionnons alors deux ensembles de données à comparer.
2. En haut à droite, nous avons des options pour coller des données à partir du presse-papiers (Coller), charger des données à partir d'un fichier (Charger), supprimer la ligne actuelle (Supprimer) et effacer tous les ensembles de données (Effacer).
3. Enfin, en bas à droite, nous avons la possibilité de comparer nos ensembles de données par mots ou par octets. Ne vous souciez pas de savoir lequel de ces boutons vous sélectionnez car cela peut être modifié ultérieurement. Ce sont les boutons sur lesquels nous cliquons lorsque nous sommes prêts à comparer les données que nous avons sélectionnées.

Comme avec la plupart des modules Burp Suite, nous pouvons également charger des données dans Comparer à partir d'autres modules en cliquant avec le bouton droit de la souris et en choisissant "Envoyer au comparateur".

Lorsque nous avons chargé des données à comparer, nous obtenons une fenêtre contextuelle nous montrant la comparaison :



Encore une fois, il y a trois parties distinctes dans cette fenêtre :

1. Les données comparées occupent la majeure partie de la fenêtre ; cela peut être visualisé au format texte ou hexadécimal. Le format initial est déterminé selon que nous avons choisi de comparer par mots ou par octets dans la fenêtre précédente, mais cela peut être écrasé en utilisant les boutons au-dessus des cases de comparaison.
2. La clé de comparaison se trouve en bas à gauche, indiquant les couleurs indiquant les données modifiées, supprimées et ajoutées entre les deux ensembles de données.
3. En bas à droite de la fenêtre se trouve la case à cocher "Synchroniser les vues". Lorsqu'il est sélectionné, cela signifie que les deux ensembles de données synchroniseront les formats - c'est-à-dire que si vous changez l'un d'eux en vue Hex, l'autre fera de même pour correspondre.

On nous donne le nombre total de différences trouvées dans le titre de la fenêtre.

4.5 Présentation Séquencer

Séquencer est l'un de ces outils rarement utilisé dans les CTF et autres environnements de laboratoire, mais il constitue un élément essentiel d'un test de pénétration d'applications Web dans le monde réel.

En bref, séquencer nous permet de mesurer l'entropie (ou le caractère aléatoire, en d'autres termes) des "tokens" - des chaînes qui sont utilisées pour identifier quelque chose et qui devraient, en théorie, être générées de manière cryptographiquement sécurisée. Par exemple, nous pouvons souhaiter analyser le caractère aléatoire d'un cookie de session ou d'une soumission de) protégeant **un jeton CSRF (Cross-Site Request)** formulaire. S'il s'avère que ces jetons ne sont pas générés de manière sécurisée, nous pouvons (en théorie) prédire les valeurs des jetons à venir. Imaginez simplement les implications de cela si le jeton en question est utilisé pour les réinitialisations de mot de passe...

Commençons, comme toujours, par jeter un œil à l'interface de séquencer :

The screenshot displays the 'Sequencer' tab of a web application. The top navigation bar includes 'Dashboard', 'Target', 'Proxy', 'Intruder', 'Repeater', 'Sequencer' (active), 'Decoder', 'Comparer', 'Logger', 'Extender', 'Project options', and 'User options'. Below this, a sub-navigation bar has 'Live capture' (active), 'Manual load', and 'Analysis options'. The main content area is divided into three sections:

- Select Live Capture Request:** Includes a help icon, a title, and instructions: "Send requests here from other tools to configure a live capture. Select the request to use, configure the other options below, then click 'Start live capture'." It features a table with columns '# ^', 'Host', and 'Request'. To the left of the table are 'Remove' and 'Clear' buttons. Below the table is a 'Start live capture' button.
- Token Location Within Response:** Includes a help icon, a title, and instructions: "Select the location in the response where the token appears." It has three radio button options: 'Cookie:', 'Form field:', and 'Custom location:'. Each has a corresponding dropdown menu. A 'Configure' button is to the right of the 'Custom location' dropdown.
- Live Capture Options:** Includes a help icon, a title, and instructions: "These settings control the engine used for making HTTP requests and harvesting tokens when performing the live capture." It contains three input fields: 'Number of threads:' (value 5), 'Throttle between requests (milliseconds):' (value 0), and a checked checkbox 'Ignore tokens whose length deviates by' (value 5) followed by 'characters'.

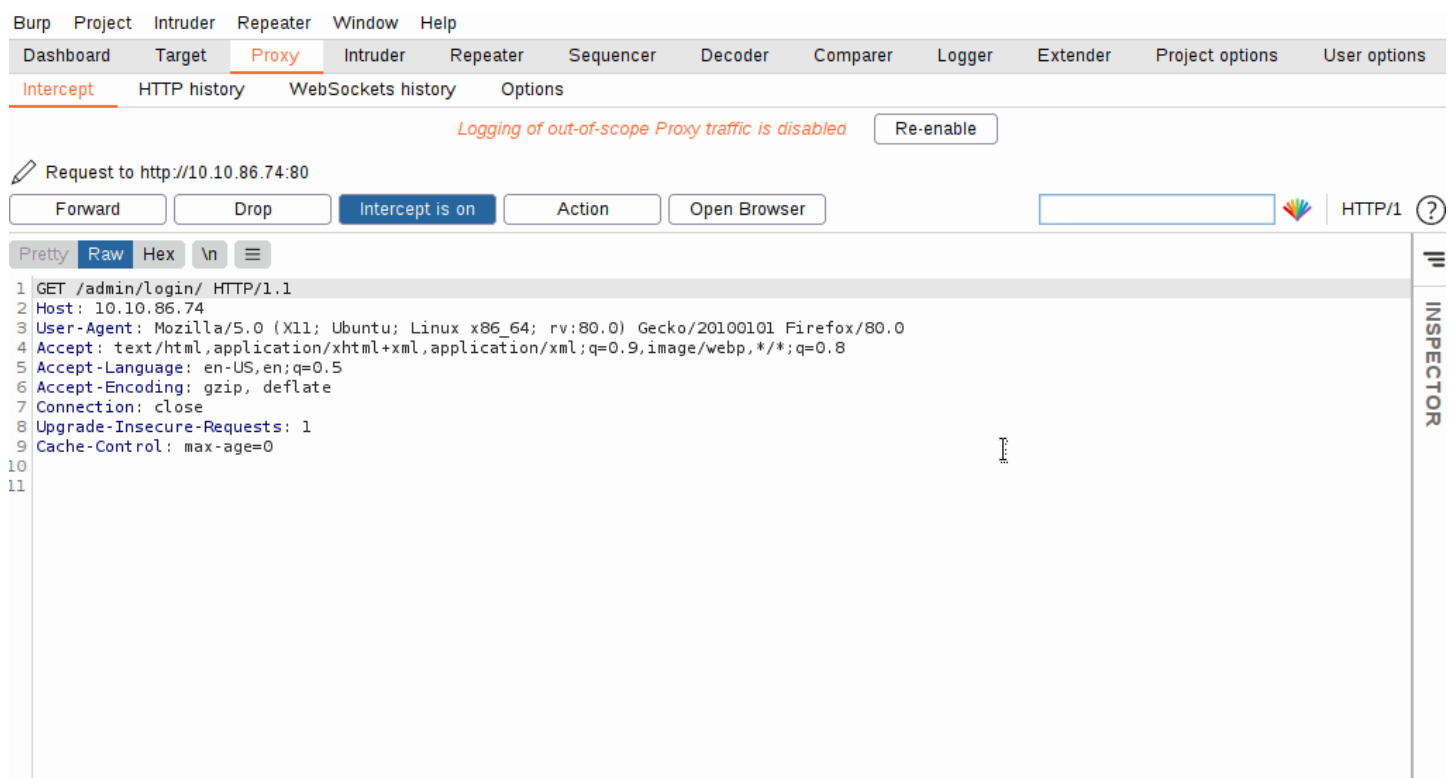
Il existe deux méthodes principales que nous pouvons utiliser pour effectuer une analyse de jeton avec séquencer :

- **La capture en direct** est la plus courante des deux méthodes - il s'agit du sous-onglet par défaut pour séquencer. La capture en direct nous permet de transmettre une demande à séquencer, qui, nous le savons, créera un jeton à analyser. Par exemple, nous pouvons souhaiter transmettre une demande POST à un point de terminaison de connexion dans séquencer, car nous savons que le serveur répondra en nous donnant un cookie. Une fois la demande transmise, nous pouvons dire à séquencer de démarrer une capture en direct : il fera alors la même demande des milliers de fois automatiquement, en stockant les échantillons de jetons générés pour analyse. Une fois que nous avons accumulé suffisamment d'échantillons, nous arrêtons séquencer et lui permettons d'analyser les jetons capturés.
- **Le chargement manuel** nous permet de charger une liste d'échantillons de jetons pré-générés directement dans séquencer pour analyse. L'utilisation du chargement manuel signifie que nous n'avons pas à faire des milliers de requêtes à notre cible (ce qui est à la fois bruyant et gourmand en ressources), mais cela signifie que nous devons obtenir une grande liste de jetons pré-générés !

4.6 Séquenceur Capture en direct

La meilleure façon d'apprendre est de faire. Apprenons à utiliser la capture en direct Sequencer en effectuant une entropie sur le jeton anti-bruteforce utilisé dans le formulaire de connexion administrateur.

Nous commencerons par capturer une demande pour `http://MACHINE_IP/admin/login/` dans la procuration. Faites un clic droit sur la requête et choisissez "Envoyer au séquenceur":



Remarquez que dans la section "Emplacement du jeton dans la réponse", nous avons la possibilité de choisir entre Cookie, Champ de formulaire et Emplacement personnalisé. Dans ce cas, nous testons le loginToken, sélectionnez donc le bouton radio pour "Champ de formulaire » :

Token Location Within Response

Select the location in the response where the token appears.

☐ Cookie:

☒ Form field:

☐ Custom location:

Nous pouvons en toute sécurité laisser toutes les autres options par défaut dans ce cas, alors allons-y et cliquez sur le bouton "Démarrer la capture en direct" !

Une nouvelle fenêtre apparaîtra maintenant nous indiquant que nous effectuons une capture en direct et nous montrant combien de jetons nous avons capturés jusqu'à présent. Nous devons attendre d'avoir un nombre raisonnable de jetons capturés (environ 10 000 devraient suffire) ; plus nous avons de jetons, plus notre analyse est précise.

Une fois que vous avez environ 10 000 jetons capturés, cliquez sur "Pause", puis sélectionnez le bouton "Analyser maintenant":

Live capture (10008 tokens)

☐ Auto analyze (next: 12000) Requests: 10008

Errors: 0

Remarque : Nous aurions également pu choisir de "Stop" la capture ; cependant, en choisissant de le mettre en pause à la place, nous laissons l'option reprendre la capture ouverte, si le rapport n'a pas assez d'échantillons pour calculer l'entropie du jeton avec précision.

Si nous voulions recevoir des mises à jour périodiques sur l'analyse, nous aurions également pu cocher la case "Analyse automatique". Faire cela indiquerait à Burp d'effectuer l'entropie toutes les 2000 requêtes ou alors, en nous donnant des mises à jour fréquentes qui seront progressivement plus précis à mesure que davantage d'échantillons sont chargés dans le séquenceur.

Il convient de noter qu'à ce stade, nous pouvons également choisir de copier ou d'enregistrer les jetons capturés pour une analyse ultérieure ultérieure.

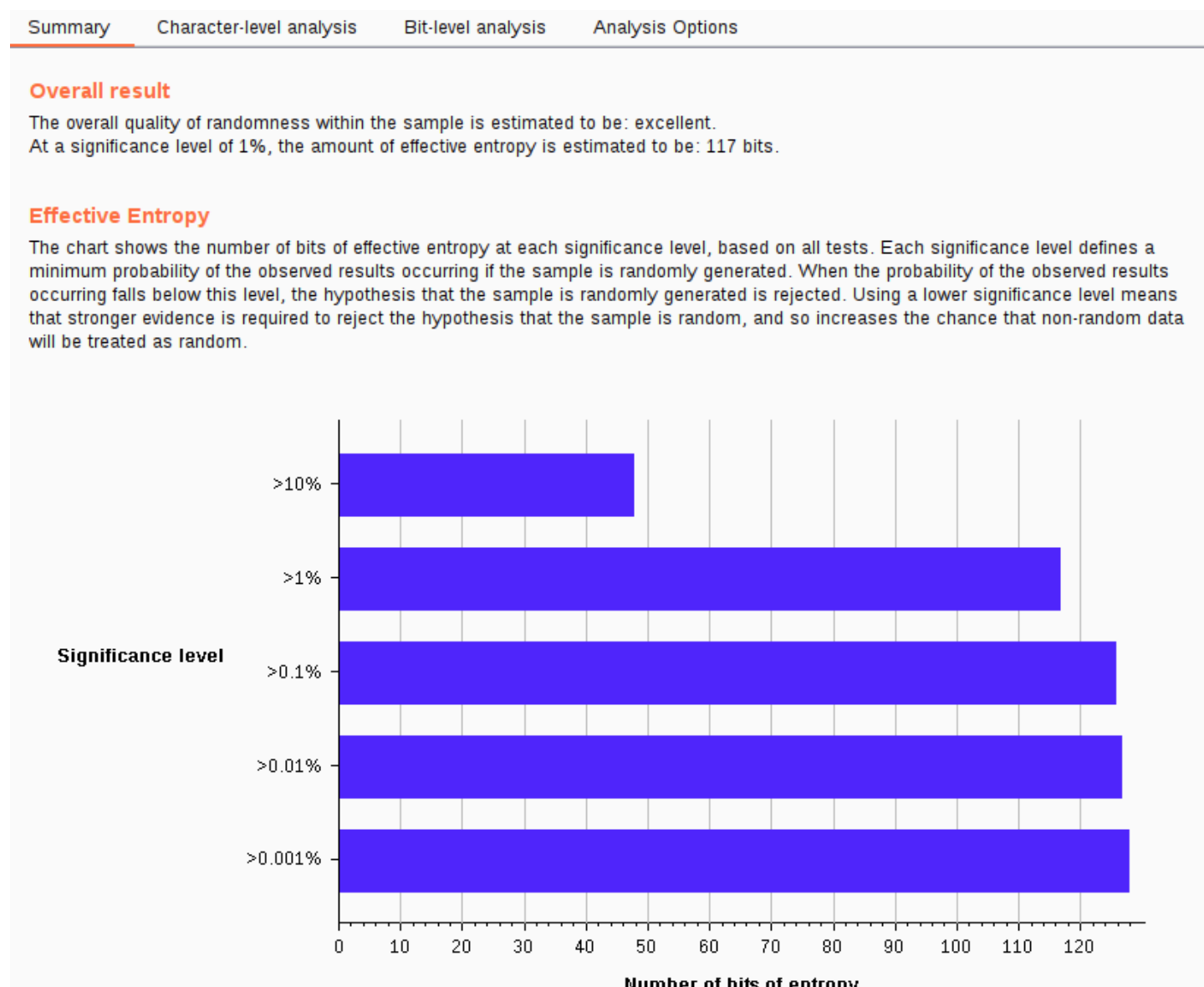
Après avoir cliqué sur le bouton "Analyser maintenant", Burp procédera à l'analyse de l'entropie de notre jeton et générera un rapport. Nous verrons cela dans la tâche suivante.

4.7 Séquencer Analyse

Maintenant que nous avons un rapport pour l'entropie de notre jeton, il est temps de l'analyser !

Burp effectue des dizaines de tests sur les échantillons de jetons qu'il a capturés. Nous n'examinerons pas tout cela car il faudrait bien plus qu'une seule tâche pour le faire (et il serait très intensif en mathématiques de séparer chaque technique). Au lieu de cela, nous nous concentrerons sur le résumé généré ; cependant, nous vous encourageons à parcourir tous les résultats des tests par vous-même.

Généré entropie est divisé en quatre sections principales, la première d'entre elles étant un résumé des résultats :



Le résumé nous donne un résultat global ; effective entropie ; une analyse de la fiabilité des résultats ; et un résumé de l'échantillon prélevé.

Collectivement, ceux-ci seront souvent suffisants pour déterminer si le jeton est généré en toute sécurité ou non ; cependant, dans certains cas, nous devons peut-être consulter directement les résultats du test - cela peut être fait dans les onglets "Analyse au niveau du caractère" et "Analyse au niveau du bit". Comme mentionné précédemment, nous n'entrerons pas dans ceux-ci pour éviter de plonger dans les profondeurs des mathématiques d'analyse statistique dans une salle adaptée aux débutants. En bref, avec une probabilité estimée à 1 % d'être incorrect sur la base des données fournies (niveau de signification : 1 %), Burp a calculé que l'entropie de notre jeton *devrait* être d'environ 117 bits. C'est un excellent niveau d'entropie à avoir dans un jeton sécurisé, même s'il convient de noter qu'il est impossible de dire avec une assurance absolue que ce calcul est entièrement exact, simplement en raison de la nature du sujet.

5. Burp Suite : Extender

5.1 L'interface de l'extendeur

Commençons par jeter un œil à l'interface de l'extendeur :

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger **Extender** Project options User options

Extensions BApp Store APIs Options

Burp Extensions ?

Extensions let you customize Burp's behavior using your own or third-party code.

Add Remove Up Down

Loaded	Type	Name
--------	------	------

...

Details Output Errors

☐ Extension loaded ?

Name:

Item	Detail
------	--------

La vue par défaut dans l'interface Extender nous donne un aperçu des extensions que nous avons chargées dans Burp Suite. Il n'y en a pas dans la capture d'écran ci-dessus - nous allons changer cela dans les prochaines tâches. La première case (vers le haut de l'interface) nous fournit une liste des extensions que nous avons installées et nous permet de les activer ou de les désactiver pour ce projet.

Les options à gauche de cette case nous permettent de désinstaller les extensions avec le bouton Supprimer ou d'en installer de nouvelles à partir de fichiers sur notre disque avec le bouton Ajouter. Il peut s'agir de modules que nous avons codés ou de modules qui ont été mis à disposition sur Internet mais qui ne se trouvent pas dans la boutique BApp. Les boutons Haut et Bas de cette section contrôlent l'ordre dans lequel les extensions installées sont répertoriées. Les extensions sont appelées dans *décroissant* fonction de cette liste. En d'autres termes : tout le trafic passant par Burp Suite sera acheminé par chaque extension dans l'ordre, en commençant par le haut de la liste et en descendant. Cela peut être très important lorsqu'il s'agit d'extensions qui modifient les demandes, car certaines peuvent se contrecarrer ou se gêner d'une autre manière.

Vers le bas de la fenêtre, nous avons les détails, la sortie et les erreurs pour le module actuellement sélectionné. Ceux-ci peuvent être utilisés pour afficher les informations du module, ainsi que pour le débogage.

5.2 Le BApp Store

Le Burp App Store (ou BApp Store en abrégé) nous permet de répertorier facilement les extensions officielles et de les intégrer de manière transparente à Burp Suite. Les extensions peuvent être écrites dans une variété de langages - le plus souvent Java (qui s'intègre automatiquement dans le framework) ou Python (qui nécessite l'interpréteur Jython - plus d'informations à ce sujet dans la tâche suivante !).

Commençons par installer une extension Java, juste pour avoir une idée de la boutique BApp.

L' [Request Timer](#) (écrite par Nick Taylor) nous permet d'enregistrer le temps que prend chaque requête que nous envoyons pour recevoir une réponse ; cela peut être extrêmement utile pour découvrir la présence (et exploiter) des vulnérabilités temporelles. Par exemple, si un formulaire de connexion prend une seconde de plus pour traiter les demandes contenant un nom d'utilisateur valide que pour les comptes qui n'existent pas, nous pouvons rapidement générer une liste des noms d'utilisateur possibles et utiliser la différence de temps pour voir quels noms d'utilisateur sont valide.

Passez au sous-onglet "BApp Store", puis recherchez "Request Timer". Il ne devrait y avoir qu'un seul résultat. Cliquez sur l'extension renvoyée, puis cliquez sur "Installer":

The screenshot shows the Burp Suite interface with the 'Extender' tab selected in the top navigation bar. Below the navigation bar, the 'Extensions' sub-tab is active. The 'Burp Extensions' section is visible, with a description: 'Extensions let you customize Burp's behavior using your own or third-party code.' Below this, there is a table with columns 'Loaded', 'Type', and 'Name'. To the left of the table are buttons for 'Add', 'Remove', 'Up', and 'Down'. Below the table, there are tabs for 'Details', 'Output', and 'Errors'. The 'Details' tab is selected, showing a checkbox for 'Extension loaded' and a text input field for 'Name:'. Below this, there is a table with columns 'Item' and 'Detail'.

Notez qu'un nouvel onglet est apparu dans le menu principal en haut de l'écran. Différentes extensions ont des comportements différents : certaines ajoutent simplement un nouvel élément aux menus contextuels du clic droit ; d'autres créent des onglets entièrement nouveaux dans la barre de menu principale.

Comme il ne s'agissait que d'un exemple d'utilisation du magasin BApp, nous n'aborderons pas ici l'utilisation de Request Timer ; cependant, il est fortement recommandé de passer au nouvel onglet et d'y jeter un coup d'œil !

[Sommaire](#)

5.3 Jython

Si nous voulons utiliser des modules Python dans Burp Suite, nous devons avoir téléchargé et inclus le fichier JAR Jython Interpreter séparé. L'interpréteur Jython est une implémentation Java de Python. Le site Web nous donne la possibilité d'installer Jython sur notre système ou de le télécharger en tant qu'archive Java autonome (JAR). Nous en avons besoin en tant qu'archive autonome pour l'intégrer à Burp.

Remarque : nous pouvons faire la même chose avec les modules Ruby et l'intégration JRuby ; cependant, nous ne couvrirons pas cela ici car : A) les modules Python sont beaucoup plus courants et B) c'est exactement le même processus pour les deux.

Tout d'abord, nous devons télécharger une copie à jour de l'archive Jython JAR à partir du [site Web de Jython](#). Nous recherchons l' **Jython Standalone** option

Enregistrez le fichier JAR quelque part sur votre disque, puis passez au sous-onglet "Options" dans Extender.

Faites défiler jusqu'à la section "Environnement Python" et définissez "Emplacement du fichier JAR autonome Jython" sur le chemin de l'archive :

Current Version

The current version of Jython is 2.7.2 It can be downloaded here:

- [Jython Installer](#) - Use this to install Jython. ([metadata](#))
- [Jython Standalone](#) - Use this to run Jython without installing or to embed Jython in a Java application. ([metadata](#))
- You may cite Jython 2.7.2 as a [dependency in your Maven or Gradle build](#).

For information on installing see [Installation](#).

This version is supported on Java 8 (minimum) and 11.

Dashboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Extender	Project options	User options
Extensions	BApp Store	APIs	Options							
<h3>Settings</h3> <p>These settings control how Burp handles extensions on startup.</p> <ul style="list-style-type: none"><input checked="" type="checkbox"/> Automatically reload extensions on startup<input checked="" type="checkbox"/> Automatically update installed BApps on startup										
<h3>Java Environment</h3> <p>These settings let you configure the environment for executing extensions that are written in Java. If your extensions use any libraries, you can specify a folder from which libraries will be loaded.</p> <p>Folder for loading library JAR files (optional):</p> <div><input type="text"/></div> <div>Select folder ...</div>										
<h3>Python Environment</h3> <p>These settings let you configure the environment for executing extensions that are written in Python. To use Python extensions, you will need to download Jython, which is a Python interpreter implemented in Java.</p> <div><p>Location of Jython standalone JAR file:</p><div><input type="text" value="/opt/burpsuite/jython-standalone-2.7.2.jar"/></div><div>Select file ...</div></div> <p>Folder for loading modules (optional):</p> <div><input type="text"/></div> <div>Select folder ...</div>										

Aussi simple que cela, nous pouvons maintenant installer des modules Python depuis le magasin BApp !

Il s'agit d'une étape très simple qui augmente considérablement le nombre d'extensions à notre disposition.

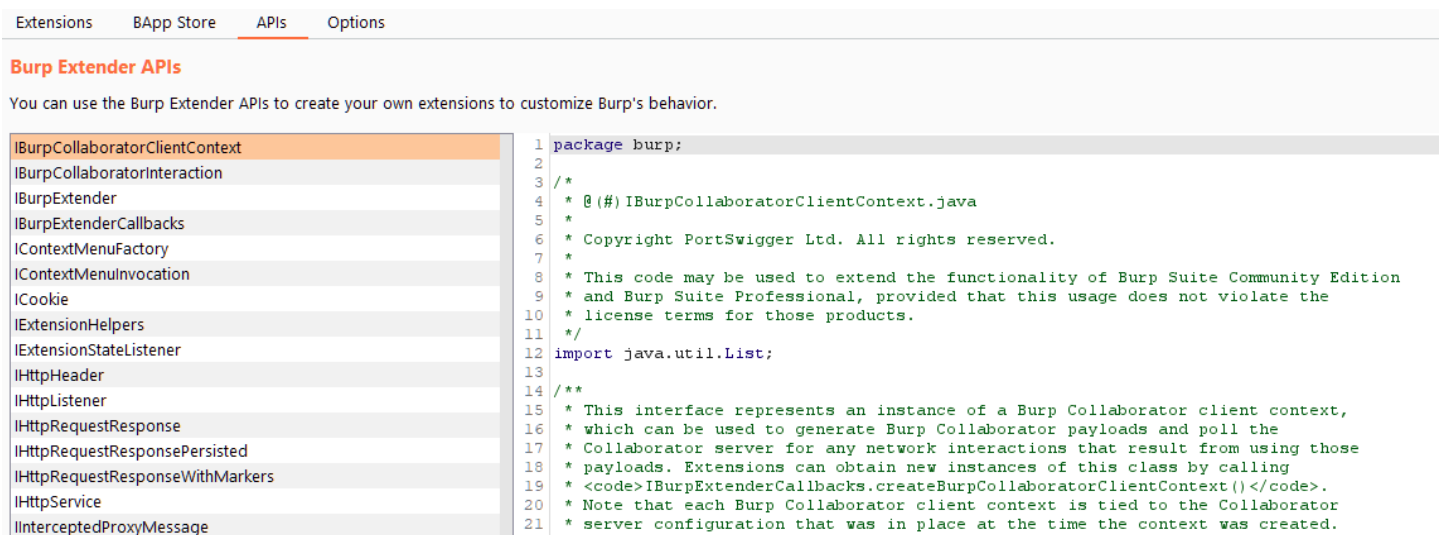
Remarque : En raison de la nature multiplateforme de Java, les mêmes étapes fonctionneront exactement pour ajouter Jython à Burp Suite sur n'importe quel système d'exploitation.

5.4 L'api Burp Suite

Bien que le codage de nos propres modules dépasse de loin la portée de ce module, il vaut la peine d'examiner (très brièvement) comment une telle tâche pourrait être abordée.

Extender expose un grand nombre de points de terminaison API auxquels les nouveaux modules peuvent se connecter lors de l'intégration à Burp Suite.

Nous pouvons les voir dans le sous-onglet "APIs" :



The screenshot shows the 'APIs' tab in Burp Suite. On the left, there is a list of APIs including `IBurpCollaboratorClientContext`, `IBurpCollaboratorInteraction`, `IBurpExtender`, `IBurpExtenderCallbacks`, `IContextMenuFactory`, `IContextMenuInvocation`, `ICookie`, `IExtensionHelpers`, `IExtensionStateListener`, `IHttpRequest`, `IHttpRequestResponse`, `IHttpRequestResponsePersisted`, `IHttpRequestResponseWithMarkers`, `IHttpService`, and `IInterceptedProxyMessage`. On the right, a code editor shows a Java snippet for `IBurpCollaboratorClientContext`, including package declarations, imports, and comments about its usage and licensing.

Chaque élément de la liste à gauche de ce sous-onglet documente un point de terminaison d'API différent, qui peuvent tous être appelés à partir des extensions. Les points de terminaison ici donnent aux développeurs beaucoup de puissance lors de l'écriture d'extensions pour interagir de manière transparente avec les fonctionnalités existantes de Burp Suite. Comme vous vous en doutez, nous pouvons interagir avec ces points de terminaison dans n'importe lequel des langages pris en charge par Burp Suite pour une utilisation dans les extensions : Java (en mode natif), Python (via Jython) et Ruby (via JRuby).

Si vous êtes particulièrement intéressé par le codage de vos propres extensions pour Burp Suite, PortSwigger fournit une merveilleuse référence qui peut être trouvée [ici](#).